PRO-
KURSSI

HTML
OHJELMOINTI

Sami Köykkä

Suomen Atk-kustannus

HTML-ohjelmointi – PRO-kurssi

Kaikki oikeudet pidätetään. Tämän julkaisun jäljentäminen ilman tekijän kirjallista lupaa painamalla, monistamalla, äänittämällä tai muulla tavoin on tekijänoikeuslain mukaisesti kielletty.

Kaikki mahdollinen on tehty kirjan tietojen oikeellisuuden varmistamiseksi. Suomen Atk-kustannus Oy ei vastaa kirjan tarkkuudesta ja kattavuudesta eikä niistä aiheutuvista kustannuksista.

Copyright Suomen Atk-kustannus Oy 1997/1998

Kirjoittajat	Sami Köykkä
Kustantaja	Suomen Atk-kustannus Oy PL 68 02601 ESPOO Puhelin: 09-4765 8550 Telekopio: 09-5765 8595 Sähköposti: palvelu@satku.fi Internet: www.satku.fi
Kansi	Frank Chaumont / Timo Ravela
Taitto	Tekijä
ISBN	951-762-477-8
Sähköinen laitos	1. Sähköinen laitos 1998 Helsinki Painettu laitos Jyväskylä 1997

HTM-ohjelmointi Pro-kurssi

SISÄLLYSLUETTELO

HUOMAA: LÖYDÄT NOPEIMMIN TIETOA KÄYTTÄMÄLLÄ ACROBAT SEARCHIN VAPAATA TEKSTIHAKUA!

Johdanto 1

Tervetuloa WWW-sivujen suunnittelijaksi 2

Miksi opettelisin HTML-kieltä? 3

Onko sivujen suunnittelu vaikeaa? 4

Miksi tekisin WWW-sivuja? 4

Kirjasta 5

Ennen sivunsuunnittelun aloitusta 7

Sivujensuunnittelun työkalut 8

Työympäristö 9

Selain 9

HTML-editori 12

Kuvankäsittelyohjelma 17

FTP 19

Muita apuohjelmia 19

Selaimen käyttö 20

Tekstieditorin käyttö 22

Ensimmäinen WWW-sivusi 27

Tekstipohjainen kotisivu 28

Mikä on HTML? 28

Leipäteksti 29

Otsikot 33
Katselu selaimessa 35
Lisää HTML-komentoja 39
Kuvitettu kotisivu 50
Kuvankäsittelyohjelma 50
Kuvat WWW-sivulla 55
Kuvien kanssa työskentely 61
Kuvien sijoittelu 61
Kuvan linkittäminen muualta verkosta 66
Kuvien kopioiminen verkosta 70
Kuvien tekijänoikeudet 70
Linkit - tiedostojen liittäminen toisiinsa 72
Linkit verkkoon 74
Kuvalinkit 75
Näkymättömiä mutta tärkeitä 77
Kommentointi 77
Sivun rakenne 78
Mallia muiden sivuista 83
Kotisivu verkkoon - sivun päivittäminen palvelimelle 85
FTP:n käyttö 86
Kotisivusi osoite 91
Etusivutiedosto 93
Operaattorien ohjesivuja 94
Lukuoikeuksien päivittäminen 95
Tiedostojen kanssa työskentely 98

Sivunsuunnittelun jatkokurssi 101

Layout - sivun ulkoasu 102
Värit 102
Taustakuvat 109
Fontit 111
Taulukot 115

Erikoiskappaleet 135
Kuvankäsittely 143
Kuvaformaatit 143
Läpinäkyvät värit 146
Vähitellen näkyviin tuleva kuva 149
Värimäärät 150
GIF-animaatio 152
Interaktiiviset kuvat 158
Postimerkkikuvat 165
Tiedostojen käsittely 167
Hakemistot 167
Hakemistot ja URL-osoitteet 169
Tiedostonnimen tarkentimet 173
Lomakkeet 175
Lomakealue 175
Ensimmäinen lomakkeesi 176
Lisää tekstikentistä 179
Painikkeet 181
Valikot 185
HTML-versiot 188
Sivukokonaisuuksien suunnittelu 189
Web-sivujen tarkoitus 189
Rakenne 190
Hyvä käyttöliittymä 204
Ylläpito 211
Intranet-suunnittelu 219
Kun sivut ovat valmiita - julkistaminen 221
Suomalaisia uutuuslistoja 221
Ulkomaisia linkkilistoja 222
Hakukoneet 222
Uutisryhmät 223

Edistynyt sivujensuunnittelu 225

Kehykset 226
Kehyksien ominaisuudet 233
Linkkien toiminta kehyksissä 234
Tyylipohjat 236
Tyylipohjien komennot 236
Tyylitiedostot 238
Tyylimäärittelyt HTML-tiedostossa 240
Tyylimäärittelyt komennoissa 242
Tyyliluokat 244
Tyylimäärittelyjen komennot 246
Tekstityypit 246
Värit 247
Marginaalit ja tekstin sijoittelu 248
Tasot 249
Tasot HTML-komennoilla 249
Tasot tyylimäärittelyiden avulla 259
Lisätietoja 262
Layout-trikkejä 263
Kuvien paloittelu ja yhdistäminen 263
Pikselikuvat 267
Multimedia sivuilla 271
Ääni 271
Videopätkät 272
Yhdistetty multimedia 274
Imuroitavat ohjelmat 275
JavaScript 276
JavaScript-perusteet 276
Muuttujat 279
Kontrollikomennot 283
Funktiot 286
Oliot 288

Sisäänrakennetut oliot ... 290
Oliot ja WWW-sivut ... 293
Document-olio ... 293
Window-olio ... 297
Location-olio ... 300
History-olio ... 301
Navigator-olio ... 302
Tapahtumat ... 303
JavaScript ja tasot ... 306
Lisätietoja ... 310
CGI-ohjelmat ... 311
Lomakkeiden käsittely ... 315
WWW-sivujen filosofiaa ... 318
Tehokkaiden ja hyvien sivujen suunnittelu ... 318
Kysymyksiä sivujen suunnittelijoille ... 322

Liitteet ... 327
Sanasto ... 328
HTML-komennot ... 341
Merkkikoodit ... 378
Lähteet ja linkit ... 384
Ohjelmat ... 388

Hakemisto ... 389

HUOMAA: LÖYDÄT NOPEIMMIN TIETOA KÄYTTÄMÄLLÄ ACROBAT SEARCHIN VAPAATA TEKSTIHAKUA!

JOHDANTO

TERVETULOA WWW-SIVUJEN SUUNNITTELIJAKSI

Tervetuloa WWW-sivujen suunnittelun ihmeelliseen maailmaan! Tämän kirjan avulla opit tekemään WWW-sivuja alusta loppuun asti. Opit tekemään itsellesi kotisivun, julkaisemaan sen kaikkien Internet-käyttäjien luettavaksi ja muokkaamaan sivuistasi tyylikkäät ja helppokäyttöiset. Kirja on tarkoitettu kaikille WWW-sivujen tekemisestä kiinnostuneille, niin aloittelijoille kuin pidemmälle ehtineillekin.

Kirjassa lähdetään liikkeelle perusasioista, joten voit tutustua WWW-suunnitteluun, vaikka et olisikaan tietokone-alan asiantuntija. Ainoat perustiedot ja -taidot joita tarvitset ovat tietokoneesi peruskäyttötaidot (osaat käyttää hiirtä ja käynnistää ohjelmia). Ennen kaikkea tarvitset avointa ja oppivaista mieltä. Oletan myös, että olet tutustunut Internetin käyttöön jo ainakin pari kertaa ennenkin. Loput asiat opit matkan varrella!

Opetettavat asiat eivät kuitenkaan jää perusasioihin vaan opit kaikki viimeisimmät HTML-komennot ja tekniikat, joita käytetään nykyaikaisten WWW-palveluiden rakentamisessa.

Monet uudet ohjelmistot vaativat paljon koneeltasi. WWW-sivuja suunnitellessa näin ei ole kuitenkaan asioiden laita: voit rakentaa itsellesi sivuja vanhemmillakin koneilla. Ainoana vaatimuksena on se, että koneeseesi löytyy tekstieditori ja WWW-selain. Internet-yhteys on myös hyvä olla, mutta sen puuttuminen ei estä tämän kirjan hyödyntämistä. Kirjan esimerkit on tehty Windows 95:llä ja Netscapella, mutta voit käyttää samoja esimerkkejä lähes muuttamatta melkein missä tahansa kotitietokoneessa.

Kirja jakautuu neljään osaan:

Ensimmäisessä osassa tutustutaan käytettäviin ohjelmiin ja harjoitellaan niiden käyttöä. Tutustut selaimiin ja erilaisiin editoreihin sekä opit, mitä muita ohjelmia tarvitaan sivuja tehtäessä.

Toisessa osassa rakennetaan ensimmäinen kotisivu. Tässä osassa opetellaan myös kuvankäsittelyohjelman toiminta ja piirretään sen avulla kuvia kotisivulle. Opit myös linkkien tekemisen ja luvun lopussa tiedostojen julkaisemisen Internetissä.

Kolmannessa osassa tutustutaan paremmin HTML-kieleen ja sivujen tekemiseen, opetellaan taiton suunnittelua, lomakkeiden tekemistä, kuvien käsittelyä ja tiedostorakenteen suunnittelua. Lisäksi tutustutaan sivunsuunnittelun yleisempiin periaatteisiin ja julkistetaan valmis sivu verkossa oleville linkkilistoille.

Neljännessä osassa käsitellään pitkälle vietyjä sivunsuunnittelutekniikoita: kehyksiä, tyylitiedostoja, tasoja, JavaScript- ja CGI-ohjelmointia ja tutustutaan WWW-sivukokonaisuuksien suunnitteluun vielä syvällisemmin.

MIKSI OPETTELISIN HTML-KIELTÄ?

Sivujen suunnittelussa ja tekemisessä lähdetään liikkeelle ruohonjuuritasolta: komentoihin tutustutaan yksi kerrallaan esimerkkien avulla ja ohjelmina käytetään yksinkertaisimpia tekstieditoreita. Jos olet tutustunut aiemmin kotisivun tekemiseen, sinulle saattaa tulla mieleen kysymys "Miksi opettelisin HTML-kieltä, koska nykyään on tarjolla niin monipuolisia apuohjelmia?".

HTML-kielen opiskelu ei ole kuitenkaan turhaa, sillä parhaimmatkin editorit ovat vain apuvälineitä. Samalla kun opettelet HTML-komentoja ja erilaisia tekniikoita, opit tuntemaan WWW-sivujen mahdollisuudet ja rajoitukset. Voit käyttää sivujen suunnittelussa keinoja, joita kaikissa editoreissa ei ole tarjolla. Voit myös vaihtaa vapaasti käyttämäsi apuohjelman uuteen ja olet välittömästi tuttu uuden ohjelman kanssa, sillä tiedät sivujesi alla piilevien komentojen ja toimintojen merkityksen.

HTML-kielen opetteleminen ei sulje sinulta pois sitä mahdollisuutta, että tekisit sivusi nykyaikaisilla suunnittelutyökaluilla - päinvastoin! Kun suunnittelet sivuja, voit käyttää sivujen rakentamiseen mitä tahansa suunnittelutyökalua. Kun törmäät ohjelman rajoituksiin tai asioihin, joita parhaimmallakaan editorilla ei voi tehdä, voit käyttää hankkimiasi taitoja hyväksesi ja muokata suoraan ohjelman tekemää HTML-tekstiä.

Voit käyttää tätä kirjaa myös suunnittelutyökalusi opaskirjana - kaikki WWW-sivujen tekemiseen tarkoitetut ohjelmat käyttävät samaa terminologiaa, joten voit tarkistaa tästä kirjasta, mitä ohjelmasi eri toiminnot tekevät.

ONKO SIVUJEN SUUNNITTELU VAIKEAA?

Kun ensimmäiset graafiset WWW-selaimet julkistettiin vuonna 1993, Internet oli tuntematon asia suurimmalle osalle ihmisistä. Siihen tuli kuitenkin muutos samalla, kun sekä verkon selailu että sisällön valmistaminen muuttuivat paljon aikaisempaa helpommiksi.

WWW-sivujen rakentaminen saattaa vaikuttaa vaikealta ja monimutkaiselta tehtävältä, jos et ole koskaan ennen tutustunut siihen. Olet ehkä nähnyt joskus tehtävän tietokoneohjelmia monimutkaisilla ohjelmointikielillä, joilla on arvoituksellisia nimiä. WWW-sivujen rakentaminen poikkeaa kuitenkin tästä. Kun olet saanut aikaan ensimmäisen sivusi, luultavasti ajattelet: näinkö helppoa tämä olikin?

Yksi WWW-sivujen tekemisen hienoista puolista on se, että HTML-kielellä on mahdollista saada aikaan näyttävää jälkeä pienellä vaivalla ja nopeasti. Jo muutaman tunnin kuluttua suunnittelet kotisivusi väritystä ja ulkoasua sekä piirrät kuvitusta sivuillesi.

Toinen jännittävä WWW:n piirre on se, että voit julkistaa työsi tulokset koko maailman ihailtavaksi todella helposti. Voit esitellä aikaansaannoksiasi parhaille tutuillesi tai saatat saada yllättäen lukijoita toiselta puolelta maailmaa. Julkaiseminen ei ole koskaan ollut näin helppoa.

MIKSI TEKISIN WWW-SIVUJA?

WWW-sivujen helpon suunnittelun ansiosta verkko on pullollaan mitä erilaisimpia hengentuotteita. Jotkut julkaisevat arvokkaita tietokokoelmia, toiset tekevät kotisivuja puhtaasti omaksi ilokseen.

Kotisivun tekemiseen voikin olla monia syitä. Haluat ehkä julkistaa maailmalle ajatuksiasi tai keräämääsi tietoa. Sarjakuvien piirtäjät saavat sarjakuvilleen välittömästi maailmanlaajuisen levikin. Opiskelijat voivat jakaa harjoitustöitään tai gradujaan lähteiksi halukkaille. Kellaribändeissä soittavat voivat esitellä uusimpia kappaleitaan.

Voit julkistaa yhteystietosi siltä varalta, että joku kaukainen tuttava tai sukulainen on hukannut puhelinnumerosi ja osoitteesi. Saatat saada sähköpostia aiemmin tuntemattomalta Amerikan-serkultasi, joka sattuu huomaamaan yhteisen sukunimen kotisivullasi.

Tai ehkäpä sinut on palkattu tekemään yrityksesi tai yhteisösi sivut. Haluatte kenties esitellä tuotteitanne verkossa tai yksinkertaisesti vain olla läsnä siltä varalta, että joku haluaa etsiä tietoa juuri omasta yhteisöstäsi. Kerhot ja seurat voivat tehdä sähköisen ilmoitustaulun jäsenilleen. Mahdollisuudet ovat rajattomat.

Kotisivun tekemiseen ei tarvita välttämättä syytä. Voit tehdä sivut puhtaasti omaksi iloksesi ja ottaa vastaan iloisena yllätyksenä sen, että joku on saapunut lukemaan sivujasi.

KIRJASTA

Olen pyrkinyt käyttämään suomenkielisinä termeinä alalla vakiintuneita termejä. Koska ohjelmistot ja itse HTML-kieli elävät jatkuvasti ja ne suunnitellaan pääasiassa englanninkielisissä maissa, kaikille termeille ei ole vielä vakiintuneita vastineita. Sen vuoksi saatat törmätä muissa lähteissä samaa asiaa tarkoittaviin erilaisiin suomennoksiin.

Itse WWW elää ja kehittyy myös jatkuvasti. Jo muutaman kuukauden päästä WWW-sivuilla saattaa olla käytössä uusia tekniikoita, joista ei tätä kirjaa kirjoitettaessa tiedetty mitään. Oppimasi asiat eivät kuitenkaan vanhennu vaan muodostavat perustan WWW-sivuille pitkälle tulevaisuuteen. Oppimasi tieto ei siis mene hukkaan.

Tämä kirja on kirjoitettu Tampereella syksyllä 1997, aikana jolloin WWW:n suosituimpia selaimia olivat Netscape 4.02 ja Internet Explorer 4.0. Kirjan teossa on ollut apuna vaimoni Riitta Santala-Köykkä, jolle haluankin välittää rakkaat kiitokset avusta!

ENNEN SIVUNSUUNNITTELUN ALOITUSTA

SIVUJENSUUNNITTELUN TYÖKALUT

Ennen sivujen tekoa kannattaa käyttää muutama hetki työkalujen hankkimiseen ja niiden käyttöön tutustumiseen. Seuraavissa luvuissa esitellään tarvitsemasi perustyökalut sekä selaimen ja editorin käyttöä.

Lisäksi tarvitset Internet-yhteyden. WWW-sivujen tekeminen on mahdollista ilman verkkoyhteyttäkin, mutta silloin et voi julkistaa sivujasi maailman luettavaksi. Jos haluat tehdä sivuja omaksi iloksesi tai vaikkapa disketeillä jaettavaksi, pärjäät mainiosti ilman Internet-liittymääkin.

Oletan, että olet käyttänyt Internetiä jo ennen kuin aloitat kotisivun tekemisen. Jos olet ensikertalainen verkossa, suosittelen että käytät ainakin yhden illan verkkoon tutustumiseen. Käynnistä rohkeasti WWW-selaimesi ja surffaile verkossa muutama tunti. Sillä tavalla tulet tutuksi sekä terminologian että verkkosurffailun kanssa yleensä. Kun hankit itsellesi Internet-yhteyden, saat todennäköisesti mukana ohjeita, jotka auttavat sinut alkuun WWW-surffailun kanssa.

TYÖYMPÄRISTÖ

HTML-sivujen tekemiseen ei tarvita monimutkaisia työkaluja. Alkuun pääsee jo pelkällä selaimella, tekstieditorilla ja kuvankäsittelyohjelmalla.

Erilaisia ohjelmia on tarjolla lukuisa määrä, sekä ilmaisia että kaupallisia ohjelmia. Voit käyttää sivujen rakentamiseen itse valitsemiasi ohjelmia - tämän kirjan ohjeet on kirjoitettu siten, että suurin osa ohjelmista soveltuu tarkoituksiisi.

Edistyessäsi sivujen suunnittelussa löydät varmasti itsellesi sopivat perustyökalut, mutta niiden lisäksi sinulle kertyy varmasti kokoelma erilaisia sivunsuunnittelun apuohjelmia. Seuraavissa kappaleissa esitellään joitain parhaista perustyökaluista. Tietoa ja vinkkejä apuohjelmista löydät myöhemmin asiaankuuluvista kohdista.

SELAIN

Selain on WWW-sivujen tekijän tärkein työkalu. Sen avulla voit katsella omia sivujasi samalla kun rakennat niitä ja sillä voit myös katsella muita Internetissä luettavana olevia WWW-sivuja.

WWW-selaimia on tarjolla useita erilaisia, mutta suurimman markkina-aseman ovat saaneet kaksi suurinta merkkiä, Netscape ja Internet Explorer. Tässä vaiheessa sinulla on todennäköisesti jo käytössäsi jonkinlainen WWW-selain, todennäköisesti jompikumpi kahdesta suurimmasta. Tarjolla on kuitenkin muitakin selainmerkkejä, joita esitellään kahden suurimman lisäksi lyhyesti seuraavissa luvuissa.

Tulevaisuudessa WWW-selaimia löytyy yhä useampia erilaisia, joskus ehkä yllättävistäkin paikoista. Jo nyt voit löytää selaimen tietokoneesi lisäksi myös erilaisista informaatiopäätteistä, tekstinkäsittelyohjelmista, televisiosta tai kännykästä.

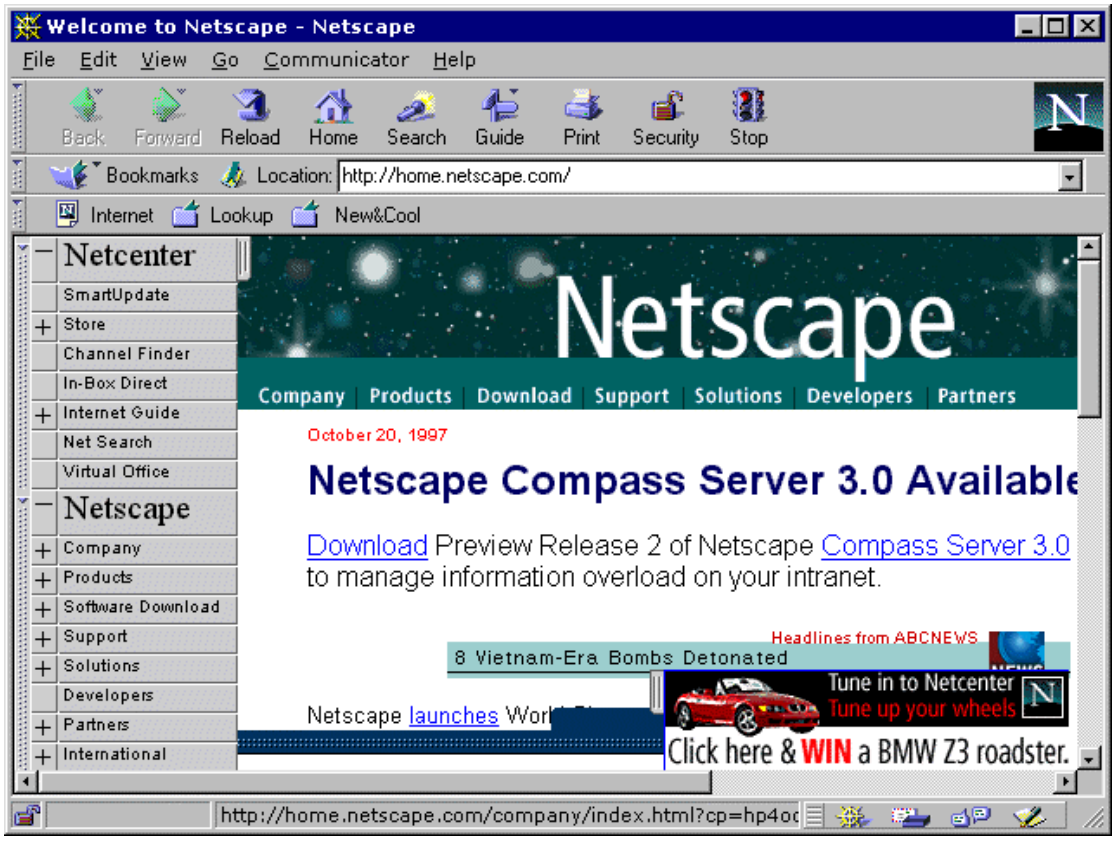
Kaikki selaimet osaavat lukea periaatteessa kaikkia WWW-sivuja. Tämän ansiosta voit käyttää sivuja katsellessasi ja suunnitellessasi useampia selainmerkkejä ja voit vaihtaa ohjelmaa koska tahansa haluat.

Yhteensopivuudessa on kuitenkin poikkeuksiakin: joillain WWW-sivuilla on käytetty toimintoja, jotka toimivat vain jossain tietyssä selaimessa. On olemassa myös selaimia, jotka näyttävät sivuja vain rajoitetussa muodossa. Esimerkiksi tekstipäätteillä tai pienissä laitteissa toimivat selaimet saattavat näyttää sivut ilman kuvia tai värejä.

Kun suunnittelet sivuja, tarkoituksenasi on varmastikin saada sivuillesi suurin mahdollinen yleisö, joten on viisasta käyttää sivunsuunnittelun aikana useampia erilaisia selaimia ja katsella tekemääsi sivua näissä. Saatat huomata, että sama sivu saattaa näyttää erilaiselta kahdessa eri selaimessa. Yksi WWW-suunnittelun suurimpia haasteita onkin tehdä sellaisia sivuja, jotka säilyvät toimivina selaimesta toiseen.

NETSCAPE

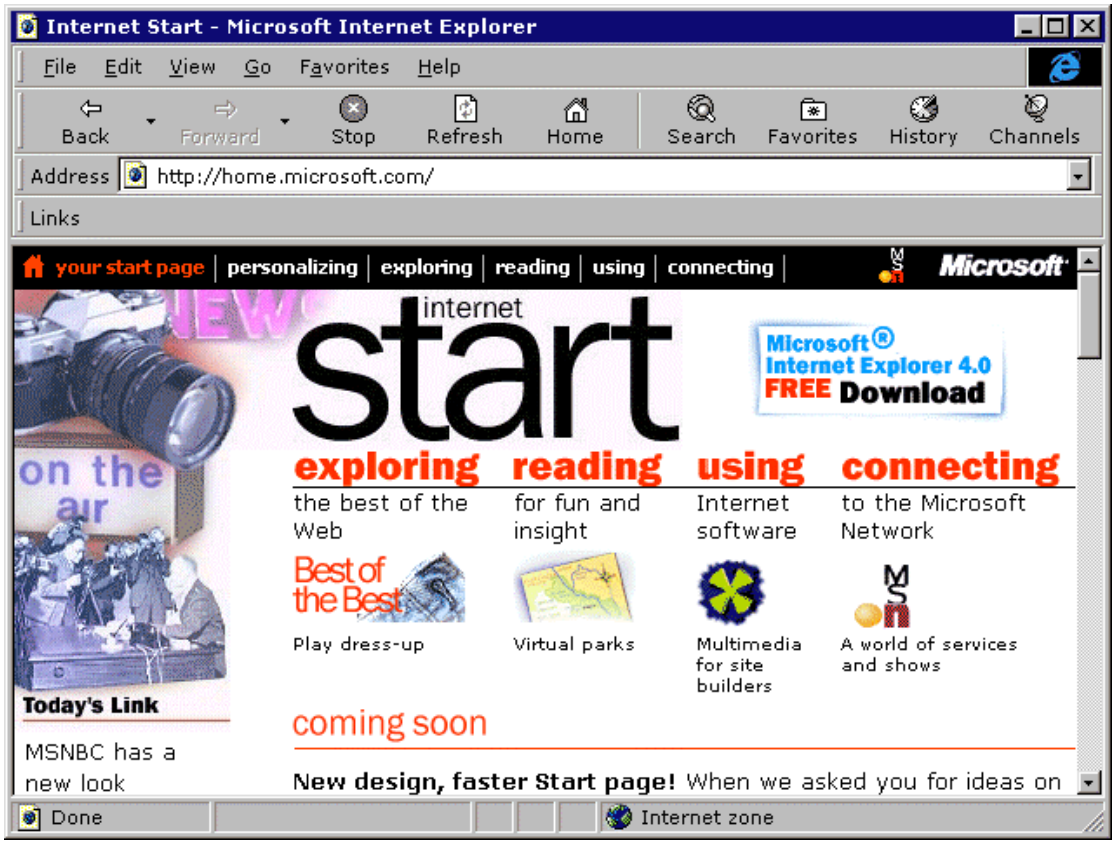
Tätä kirjoitettaessa syksyllä 1997 suosituin WWW-selain on Netscape. Netscape julkistettiin ensimmäisen kerran vuonna 1994 (ja siitä tulikin nopeasti yksi Internet-käyttäjän perustyökaluja.



Voit kopioida itsellesi Netscape-ohjelman sen kotisivulta osoitteesta **`http://www.netscape.com`**

INTERNET EXPLORER

Internet Explorer on Microsoftin valmistama kilpailija Netscapelle. Se on verkon toiseksi suosituin selainohjelma noin 30% osuudella kaikista verkkosivujen lukijoista.



Voit kopioida Internet Explorerin itsellesi maksutta Microsoftin kotisivuilta osoitteesta **`http://www.microsoft.com`**

MUUT VERSIOT

Netscape ja Internet Explorer eivät ole ainoita tarjolla olevia selaimia, vaan verkosta löytyy lukuisa määrä muitakin selaimia käytettäväksi. Myös niihin tietokoneisiin, joissa ei voi ajaa suosituimpia selaimia löytyy omia selainvaihtoehtoja.

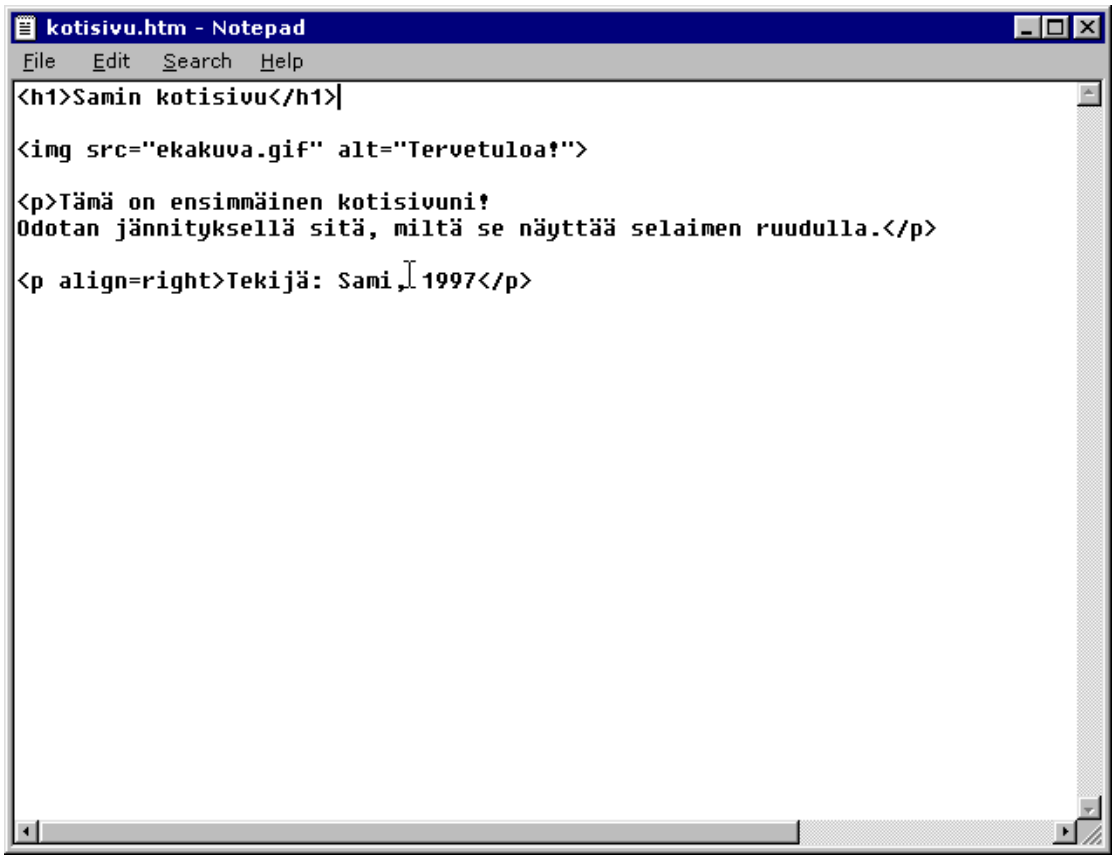
Suosittuja selaimia ovat esimerkiksi **Lynx** (tekstipäätteille tai koneille joissa ei ole graafista käyttöliittymää) ja **Opera** (Windowsille).

HTML-EDITORI

Toinen perustyökalu WWW-sivujen tekijälle on editori. Kaikki sivut muodostuvat tavallisista tekstitiedostoista, joita kirjoitetaan ja muokataan editoria apuna käyttäen.

WWW-sivujen sisältö koostuu tekstistä, jonka sisään kirjoitetaan HTML-komentoja niin ikään tavallisena tekstinä. Tämän ansiosta voit käyttää sivujen muokkaamiseen mitä tahansa ohjelmaa, jolla voi kirjoittaa ja tallentaa tekstitiedostoja.

Koska käytössä oleva ohjelmavalikoima on näin laaja, voit aloittaa sivujen tekemisen sillä ohjelmalla, jonka käytön osaat parhaiten ja joka on helpoiten saatavilla. Kaikkien tietokoneiden mukana tulee yleensä jonkinlainen editori - esimerkiksi Windowsissa ohjelma nimeltään **Notepad** ja Macintoshissa Teksturi.



Tämän kirjan esimerkit onkin tehty siten, että voit kirjoittaa HTML-sivuja tietokoneestasi löytyvällä editorilla eikä sinun tarvitse kopioida itsellesi uusia ohjelmia editointia varten.

Voit vaihtaa käyttämääsi ohjelmaa koska tahansa. Koska WWW-sivut on tallennettu koneeseesi tavallisena tekstinä, kaikki tekstinkäsittelyohjelmat ja editorit osaavat lukea niitä. Voit ottaa käyttöön nimenomaan WWW-sivujen rakentamista varten suunnitellun HTML-editorin, tekstinkäsittelyohjelman tai laajempaa sivujen rakentamista varten suunnitellun ohjelman. Voit myös vaihtaa koska tahansa käyttämääsi ohjelmaa, jos haluat kokeilla uusia ohjelmia tai jos kyllästyt vanhaan ohjelmaasi.

TEKSTIEDITORIT

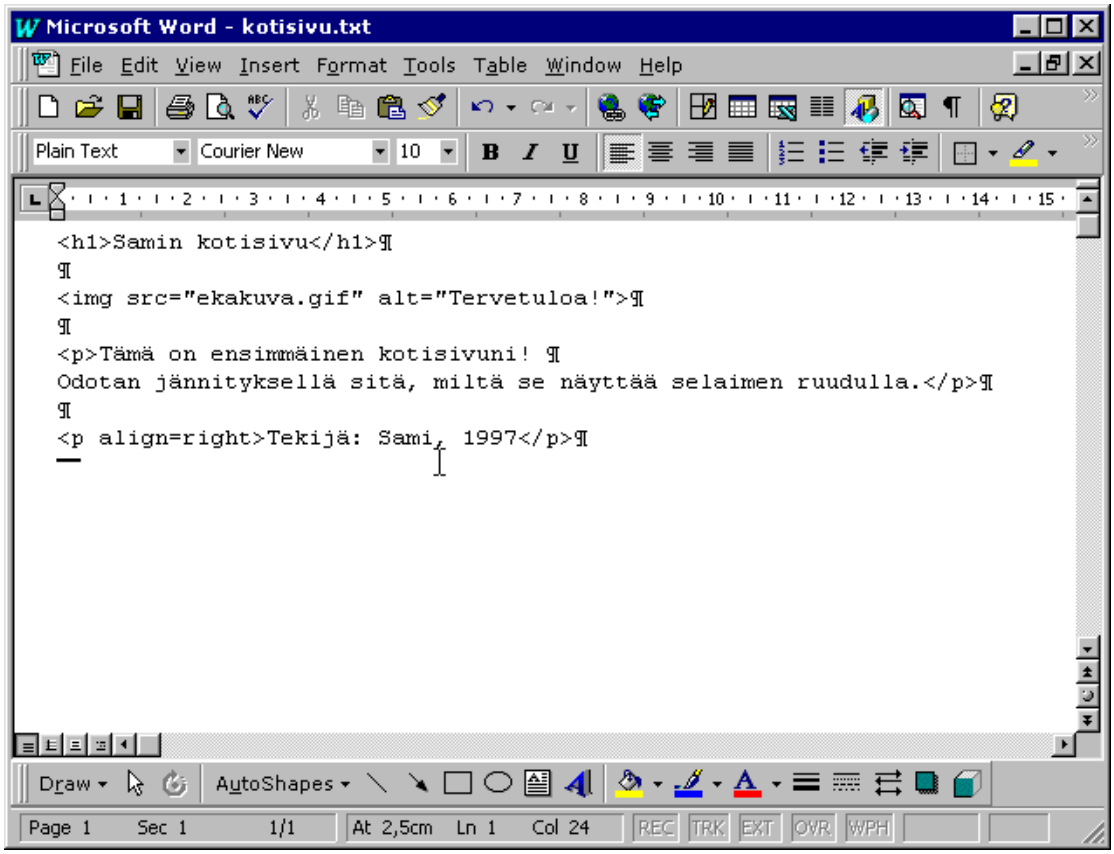
Tämän kirjan esimerkit on suunniteltu kirjoitettavaksi tietokoneesi mukana tulevalla tekstieditorilla. Esimerkkien ohjeet käsittelevät tilannetta, jossa käytät **Notepad**-ohjelmaa Windows 95:ssä, mutta samat ohjeet pätevät lähes muuttumattomina myös vanhempaan Windowsiin.

Macintosh-koneissa toimenpiteet (esimerkiksi ohjelmasta toiseen siirtyminen, ikkunoiden ulkoasu sekä ohjelmien, painikkeiden ja valikoiden komentojen nimet) saattavat erota hieman tässä kirjassa esitellyistä.

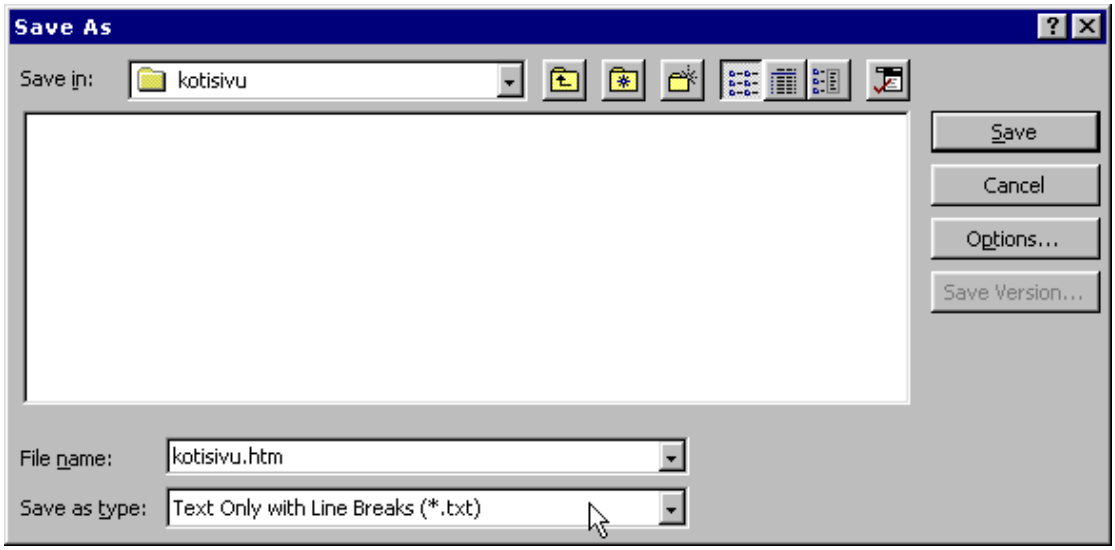
Voit käynnistää **Notepad**-ohjelman Windows 95:ssä **Start**-valikon kohdasta **`Programs/Accessories`**. Vanhemmassa Windowsissa **Notepad**in käynnistysikoni löytyy muiden apuohjelmien joukosta.

TEKSTINKÄSITTELYOHJELMAT

HTML-tiedostoja voi kirjoittaa myös kaikilla tekstinkäsittelyohjelmilla. Koska tekstinkäsittelyohjelmat lisäävät kuitenkin tiedostojen sisään erilaisia muokkauskomentoja tekstityypeille ja kappaleille, tekstinkäsittelyohjelmilla kirjoitetut HTML-tiedostot tulee tallentaa levylle useimmiten tekstimuodossa.



Kun olet kirjoittanut kirjan esimerkkejä tekstinkäsittelyohjelmalla, tallenna ne muodossa "**Text Only with Line Breaks**". Tallennusmuoto valitaan tallennuksen yhteydessä:

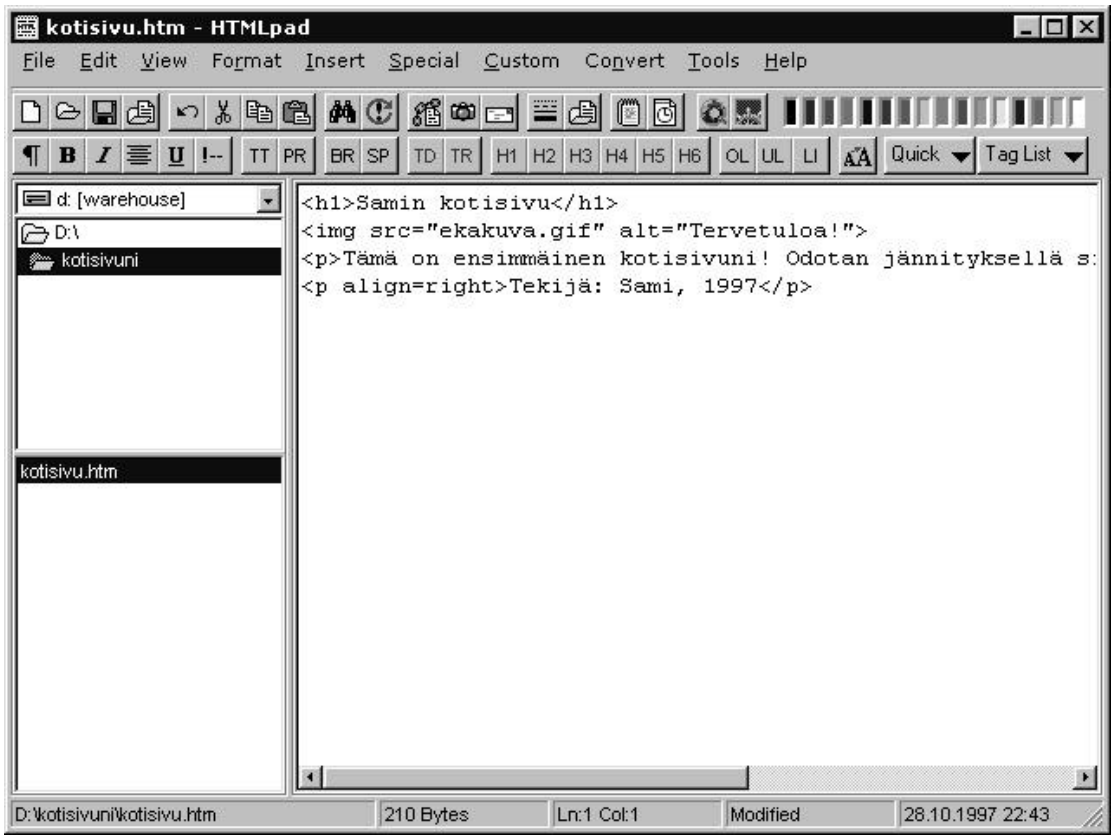


Uusimmat tekstinkäsittelyohjelmat osaavat käsitellä HTML-tiedostoja myös **WYSIWYG**-muodossa (katso lisää seuraavilta sivuilta).

HTML-EDITORIT

HTML-editorit muistuttavat toiminnaltaan tekstieditoreita. Niissäkin sivuja kirjoitetaan käyttämällä HTML-koodeja. Erona tavallisiin tekstieditoreihin on kuitenkin se, että HTML-editorit sisältävät painikkeita ja valikoita, joiden avulla komentojen lisääminen HTML-koodin sekaan helpottuu.

Esimerkki HTML-editorista on **HTMLpad**, jonka voit kopioida itsellesi sen kotisivulta osoitteesta **`http://www.intermania.com`**. **HTMLpad** muistuttaa ulkonäöltään hieman **Notepad**ia, mutta sisältää ikkunan yläreunassa useita painikkeita, joiden avulla voit lisätä sivuillesi HTML-komentoja:



Lisää HTML-editoreita löydät ohjelma-arkistoista, esimerkiksi **Tucows:ista** sivulta

`http://www.eunet.fi/tucows/htmltext95.html`

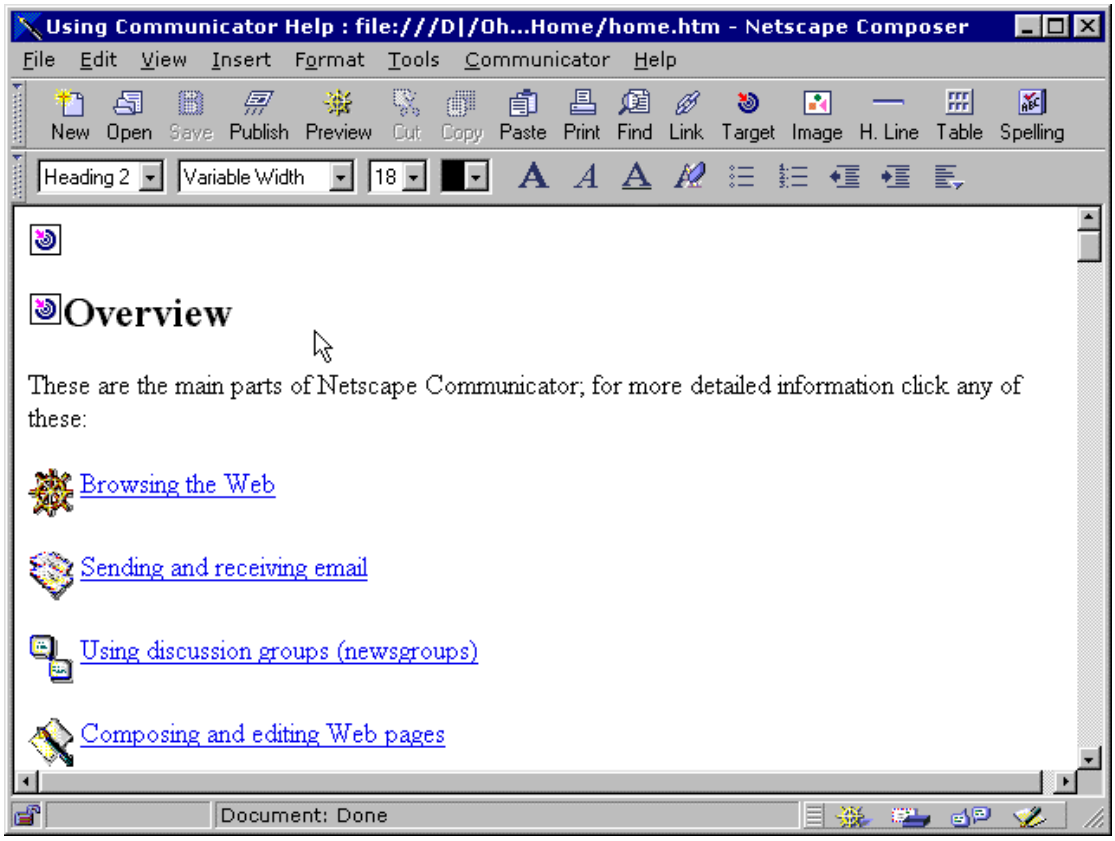
WYSIWYG-EDITORIT

Termi **WYSIWYG** tulee sanoista **What You See Is What You Get**. Tällaiset HTML-editorit näyttävät ruudulla muokkaamasi sivun miltei samanlaisena kuin miltä se tulee näyttämään selaimen ruudulla.

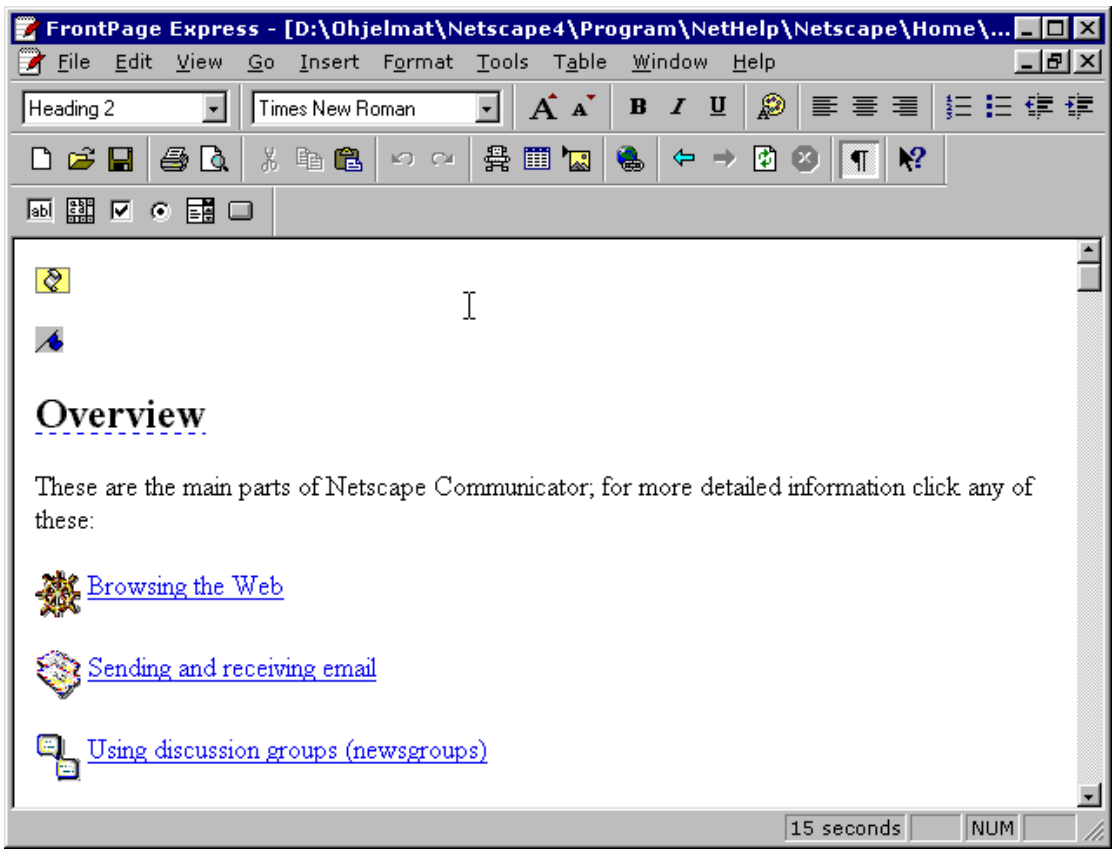
WYSIWYG-ohjelmia käyttäessäsi voit kirjoittaa WWW-sivuja samaan tyyliin kuin kirjoittaisit niitä tekstinkäsittelyohjelmalla: voit sijoittaa sivuille erilaisia tekstityylejä ja kuvia ja näet ne saman tien ruudullasi. Editori piilottaa näkyvistäsi erilaiset HTML-komennot ja näyttää sen sijasta sinulle lopputuloksen lähes sellaisena kuin se näkyy selaimen ruudulla.

Kun kirjoitat WWW-sivua, voit lisätä sivun sisään HTML-komentoja käyttämällä graafisia painikkeita ja valikoita, aivan kuten muissakin HTML-editoreissa.

Monet uusimmat WWW-selaimet sisältävät osana selainpakettia **WYSIWYG**-editorin. Netscapen nelosversion mukana tuleva ohjelma on nimeltään **Composer**:



Internet Explorerin version 4 mukana tuleva HTML-editori on nimeltään **Frontpage Express**:



WYSIWYG-editoreita on saatavilla myös kaupallisina versioina ja esimerkiksi tekstinkäsittelyohjelmien mukana. Microsoft Office 97 -pakettiin kuuluva Word-tekstinkäsittelyohjelma osaa avata HTML-tiedostot **WYSIWYG**-muodossa, jolloin voit käyttää sitä suoraan HTML-editorina.

Kaupalliset WWW-palvelimien suunnitteluohjelmistot sisältävät myös **WYSIWYG**-editorin: esimerkiksi **Microsoft Frontpage** ja **Adobe Sitemill** antavat sinun muokata WWW-sivuja suoraan graafisessa muodossa.

Voit katsella verkossa saatavilla olevien **WYSIWYG**-editoreiden valikoimaa esimerkiksi **Tucows:ista**, sivulta

`http://www.eunet.fi/tucows/htmlwysiwyg95.html`

KUVANKÄSITTELYOHJELMA

Kuvankäsittelyohjelmilla piirretään ja muokataan WWW-sivuille tulevia kuvia. Niiden avulla voit piirtää alusta asti omat kuvasi sivuillesi, tai voit retusoida muualta kopioimiasi tai skannaamiasi kuvia.

Suurin osa tarjolla olevista kuvankäsittelyohjelmista sopii WWW-grafiikan tekemiseen. Ohjelman vaatimuksena on, että se osaa käsitellä bittikarttamuodossa olevia kuvia ja että se voi tallentaa kuvat **gif**- tai **jpeg**-muodossa.

Verkosta löytyvät ohjelma-arkistot sisältävät suuren valikoiman erilaisia kuvankäsittelyohjelmia ja voitkin poimia niistä juuri itsellesi sopivan. Kaikki WWW:n kuvaformaatteja ymmärtävät kuvankäsittelyohjelmat osaavat lukea toistensa tallentamia tiedostoja, joten voit vaihtaa käyttämääsi ohjelmaa ja silti muokata myös aikaisemmin tekemiäsi kuvia.

PAINT SHOP PRO

Yksi suosituimpia kuvankäsittelyohjelmia Windowsille on **Paint Shop Pro**. Se on julkisohjelma, eli voit kopioida sen maksutta omalle koneellesi ja koekäyttää sitä 30 päivän ajan.

Paint Shop Pro sisältää kaikki WWW-grafiikan piirtämiseen ja muokkaamiseen tarvittavat perustoiminnot. Se osaa myös tallentaa ja lukea molempia WWW-sivuilla käytettäviä kuvaformaatteja, joten **Paint Shop Pro** sopii mainiosti ensimmäiseksi kuvankäsittelyohjelmaksesi.

Voit kopioida **Paint Shop Pro**n omalle koneellesi joko suoraan sen kotisivulta osoitteesta **`http://www.jasc.com`** tai monista verkon ohjelma-arkistoista.

PHOTOSHOP

Adobe Photoshop on kaupallinen ohjelma, joka on yksi suosituimpia WWW-sivunsuunnittelun ammattilaisten käyttämistä kuvankäsittelyohjelmista.

Photoshop sisältää monia hyödyllisiä kuvankäsittelytoimintoja ja sen uusimmat versiot osaavat tallentaa ja lukea WWW-sivuilla käytettäviä kuvaformaatteja. **Photoshop**in aiemmat versiot saattavat kuitenkin tarvita lisäohjelman näiden formaattien lataamiseen. Tämän ohjelman voit kopioida maksutta Adoben kotisivuilta osoitteesta **`http://www.adobe.com`**

FTP

Valmiiden HTML-sivujen julkistaminen vaatii yleensä tiedostojen kopiointia verkon yli. Tähän kopiointiin käytetään yleisesti FTP-ohjelmia (FTP = **File Transfer Protocol**), eli ohjelmia jotka on tehty tiedostojen siirtämiseen Internetiin kytkettyjen tietokoneiden välillä.

Jotkut HTML-editorit sisältävät FTP-toiminnot sisäänrakennettuina, mutta jos muokkaat sivujasi jollain muulla editorilla, tarvitset erillisen tiedostonsiirto-ohjelman.

WS-FTP JA FETCH

Jos käytät Windowsia, sopiva ohjelma tiedostonsiirtoon on **WS-FTP**. Tämän graafisen ohjelman avulla voit siirtää tiedostoja omasta koneestasi WWW-palvelimeen ja päinvastoin. Voit kopioida **WS-FTP**:n itsellesi sen kotisivulta osoitteesta

http://www.ipswitch.com/Products/WS_FTP

Macintoshilla **WS-FTP**:tä vastaava helppokäyttöinen tiedostonsiirto-ohjelma on **Fetch**. Sen löydät osoitteesta

http://www.dartmouth.edu/pages/softdev/fetch.html

MUITA APUOHJELMIA

Erilaisia WWW-sivujen tekoon tarkoitettuja ohjelmia ja apuohjelmia on runsaasti saatavilla verkossa. Voit selailla ohjelmia Internetin ohjelma-arkistoissa, joista voit myös kopioida halutessasi itsellesi näitä ohjelmia.

Suosittu ohjelma-arkisto on **Tucows** osoitteessa

http://www.tucows.com/

SELAIMEN KÄYTTÖ

Jos et ole vielä tuttu selaimesi kanssa, sinun kannattaa käyttää muutama hetki sen toimintoihin tutustumiseen. Alta löydät pikaohjeet Netscape-selaimen käyttöön. Samat toiminnot ovat käytössä myös muissa selaimissa - niiden nimet saattavat olla tosin hieman toisenlaisia.

Kaikkien selainten perustoimintoja ovat **Back**- ja **Forward**-toiminnot. **Back**-painikkeen avulla pääset peruuttamaan takaisinpäin WWW-sivulta edelliselle ruudullasi näkyneelle sivulle. **Forward** on **Back**in vastakohta: sen avulla pääset takaisin eteenpäin, jos painoit **Back**ia liian monta kertaa.

PYSÄYTYS JA PÄIVITYS

Kun Netscape lataa sivua, sen työkalupalkin oikeassa reunassa oleva ikoni näyttää animaatiota pyrstötähdistä. Tästä tiedät Netscapen tekevän juuri jotain. Voit pysäyttää latauksen painamalla **Stop**-painiketta.

Jos sivun lataus on jäänyt kesken, voit ladata sivun uudelleen painamalla **Reload**-painiketta. Samalla painikkeella voit myös ladata sivun uudelleen, esimerkiksi silloin jos tiedät että sivun sisältö on saattanut muuttua (vaikkapa uutissivuilla).

OSOITTEET

Jokaisella WWW-sivulla on oma osoite, jota kutsutaan URL:ksi (URL = **Uniform Resource Locator**). Osoitteet alkavat yleensä tekstillä **`http`**, mutta joskus saatat törmätä myös osoitteisiin, jotka alkavat tekstillä **`ftp`**, **`file`**, **`gopher`**, **`news`** tai **`mailto`**. Tämä teksti kertoo, miten selaimen tulee hakea osoitteen kuvaama sivu.

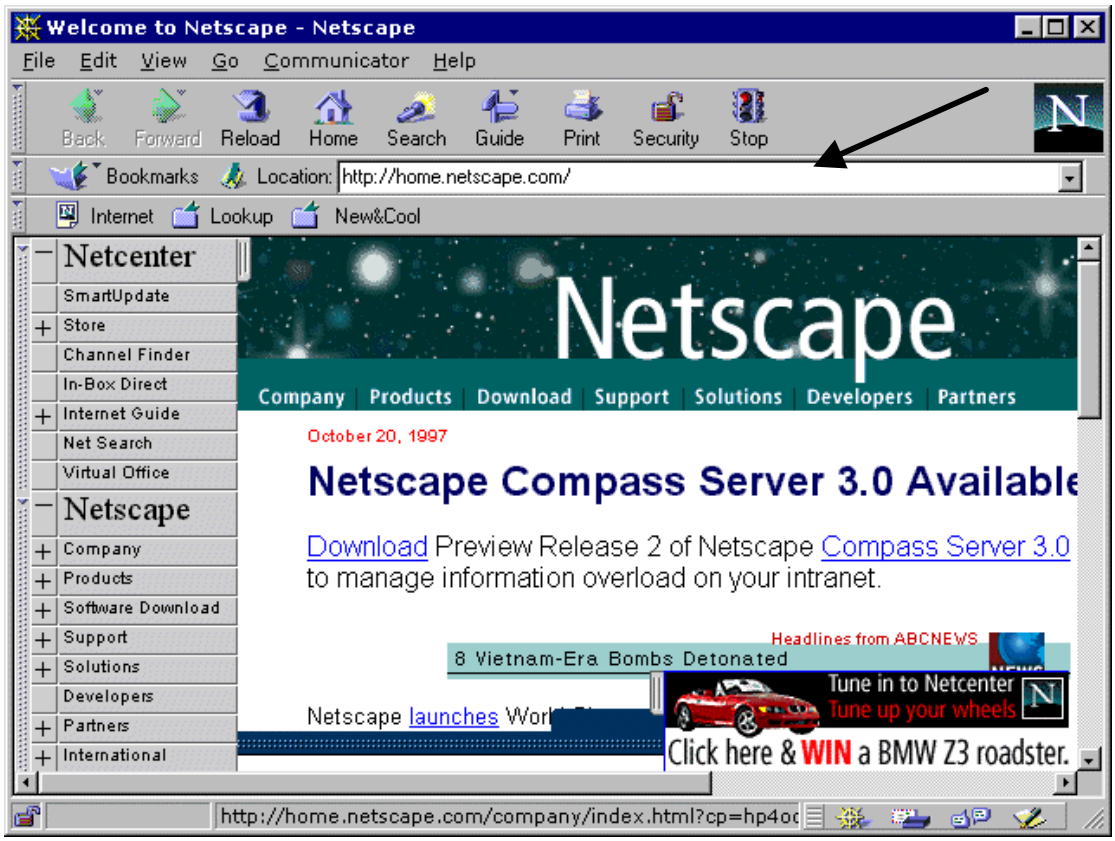
Alkutekstin jälkeen osoitteessa tulee näkyviin tietokoneen nimi, josta haluttu WWW-sivu haetaan. Koneen nimen jälkeen osoite saattaa sisältää vielä polkuja tai tiedostojen nimiä. Alla näet esimerkkejä URL-osoitteista:

`http://www.cnn.com/`
`http://www.iki.fi/~samik/`
`http://www.nlf-finland.org/projektit/index.html`

Tekstin **`http`** ja koneen nimen välissä on kaksoispiste ja kaksi kauttaviiva-merkkiä. Nämä kuuluvat osana URL-osoitteeseen.

URL-osoitteet ovat yleismaailmallisia, eli jos löydät mielenkiintoisen sivun, voit kirjoittaa itsellesi ylös sen osoitteen ja käyttää tätä osoitetta missä selaimessa tahansa, missä päin maailmaa vain.

Kun katselet selaimessa WWW-sivua, näet sen osoitteen aina selaimen osoitepalkissa. Voit kopioida osoitteen itsellesi tästä palkista.



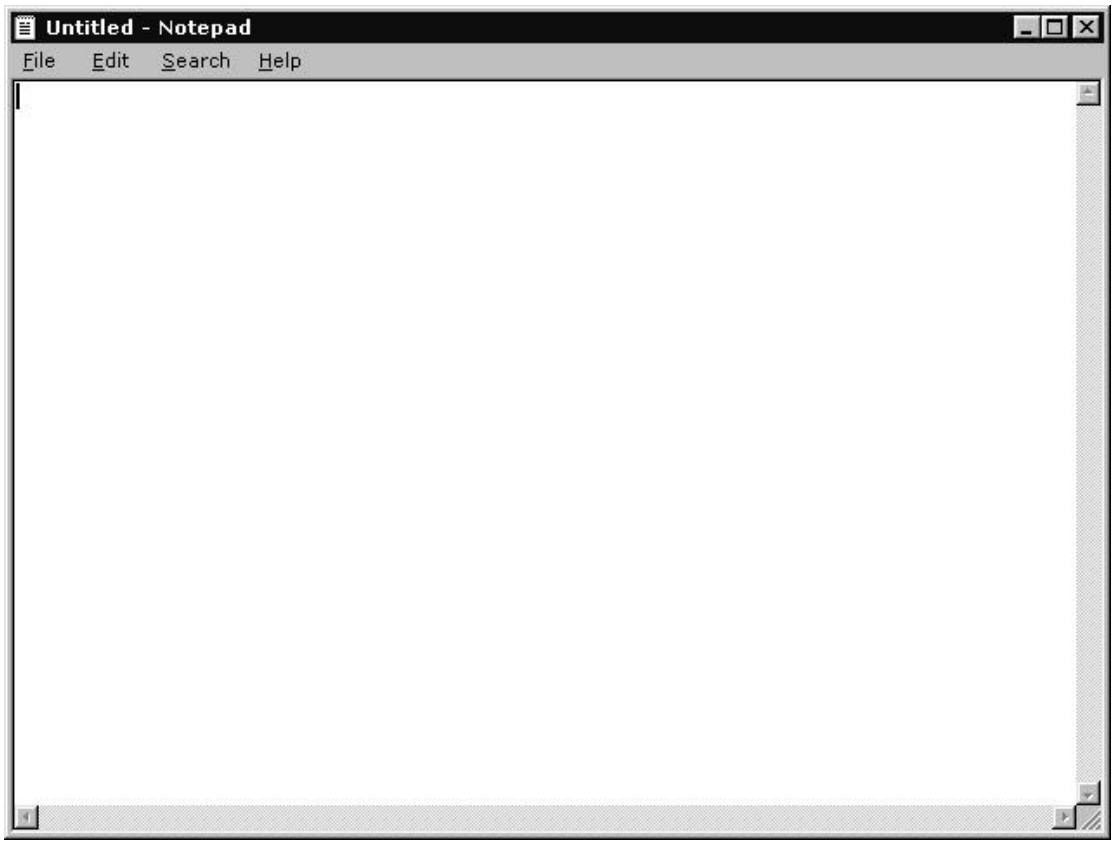
Jos haluat hypätä jollekin tietylle sivulle, voit napsauttaa hiirellä kerran osoitepalkkia, jonka jälkeen voit kirjoittaa uuden osoitteen tähän palkkiin. Kun painat Return-näppäintä, selaimesi hakee näytöllesi haluamasi sivun.

Osoitteet kirjoitetaan aina yhdelle riville, eivätkä ne sisällä välilyöntejä. Lehdissä ja kirjoissa (tässäkin!) osoitteet joudutaan joskus jakamaan tilanpuutteen vuoksi kahdelle riville, mutta ne kirjoitetaan selaimessa ja HTML-koodissa aina yhdelle riville.

TEKSTIEDITORIN KÄYTTÖ

Tämän kirjan esimerkit on suunniteltu siten, että voit käyttää sivujen rakentamiseen yksinkertaista tekstieditoria. Jokaisen tietokoneen mukana tulee jonkinlainen tekstieditori - seuraavassa esitellään miten käytät Windows 95:n **Notepad**-editoria. **Notepad** on käytössä myös Windows 3.1:ssä, jossa sitä käytetään samalla tavalla.

Windows 95:n **Notepad** käynnistyy **Start**-valikosta, kohdasta **`Programs/Accessories/Notepad`**. Kun käynnistät editorin, saat ruudullesi tyhjän **Notepad**in ikkunan:



KIRJOITTAMINEN

Voit kirjoittaa **Notepad**in ikkunaan tekstiä suoraan näppäimistöltäsi. Toisin kuin tekstinkäsittelyohjelmissa, **Notepad**issa painetaan Return-näppäintä jokaisen rivin lopussa.

Jos haluat kopioida tekstiä, voi maalata sen hiirelläsi ja valita **Edit**-valikosta toiminnon **Copy**. Tämän jälkeen maalaamasi teksti siirtyy leikepöydälle. Voit liimata sen johonkin toiseen kohtaan viemällä kursorin haluamaasi paikkaan ja valitsemalla **Edit**-valikosta toiminnon **Paste**. **Edit**-valikon toiminto **Cut** leikkaa tekstin leikepöydälle, eli tekee saman toiminnon kuin **Copy** ja lisäksi poistaa maalatun tekstin.

Leikepöydällä olevaa tekstiä voi siirtää myös eri ohjelmien välillä, eli voit kopioida tai leikata tekstiä missä tahansa ohjelmassa ja liimata sen mihin tahansa toiseen ohjelmaan.

Jos et ole käyttänyt ennen **Notepad**ia, harjoittele hetki tekstin kirjoittamista sekä leikkaamista ja liimaamista.

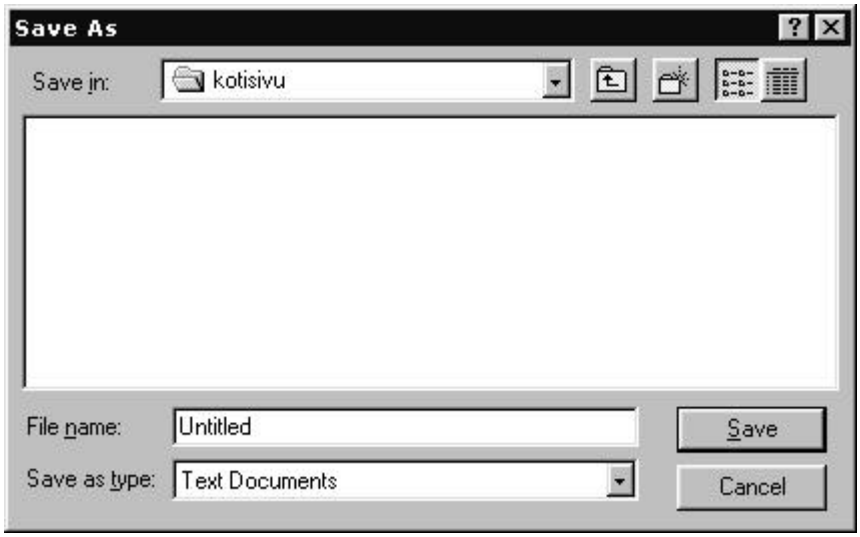
Vinkki: Voit käyttää Copy-, Cut- ja Paste-toimintojen sijasta myös näppäinoikoteitä:

Ctrl-C kopioi
Ctrl-X leikkaa
Ctrl-V liimaa

Näppäinoikotiet toimivat painamalla kahta nappia yhtäaikaa: esimerkiksi kopiointi tapahtuu painamalla samaan aikaan Ctrl- ja C-näppäimiä.

TALLENTAMINEN JA LATAAMINEN

Voit tallentaa kirjoittamasi tekstin valitsemalla **File**-valikosta toiminnon **Save**. Kun tallennat tiedostoasi ensimmäistä kertaa, **Notepad** avaa sinulle ikkunan jossa se kysyy nimeä, jolla haluat tallentaa tiedostosi.



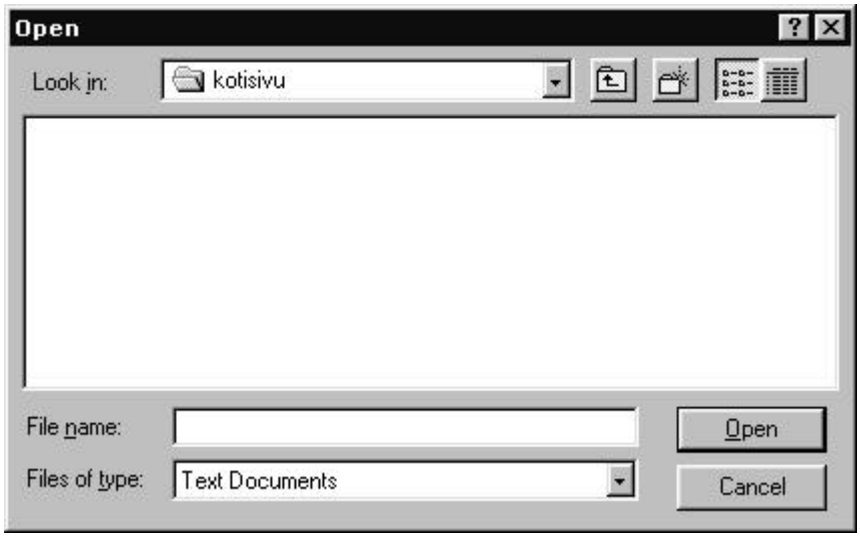
Ylhäällä olevilla painikkeilla valitaan kansio, jonne sivusi tallennetaan. Kohtaan **File name** voit kirjoittaa nimen, jolla haluat tallentaa tiedostosi.

Jos et anna tiedostollesi tarkenninta, **Notepad** lisää nimeen päätteen **`.txt`**, eli jos kirjoitat tallennettavan tiedoston nimeksi "**`kotisivu`**", **Notepad** tallentaa sen levylle nimellä "**`kotisivu.txt`**".

Voit antaa tiedostoille myös itse tarkentimen kirjoittamalla sen tiedoston nimeen. Kun kirjoitat tiedoston nimeksi "**`kotisivu.htm`**", se tallentuu levylle samalla nimellä.

Kun myöhemmin tallennat tiedoston **File**/**Save**-komennolla, **Notepad** käyttää aiemmin antamaasi tiedostonnimeä. Voit antaa tiedostolle uuden nimen **File**-valikon kohdassa **Save As**.

Voit avata aiemmin tallentamasi tiedoston **File**-valikon kohdasta **Open**. **Notepad** tuo näkyviin tiedoston latausikkunan:



Voit vaihtaa kansiota ikkunan yläreunan valikoilla aivan kuten tiedostoa tallennettaessakin. Kirjoita haluamasi tiedoston nimi kohtaan **File name**.

Keskellä olevassa listassa näet samassa hakemistossa olevat muut tiedostot. **Notepad** näyttää aluksi vain hakemistossa olevat tekstitiedostot (ne, joiden tiedostonnimen pääte on `.txt`). Voit valita valikosta "**Files of type**" kohdan "**All files**", jolloin **Notepad** näyttää ikkunassa kaikki hakemistossa olevat tiedostot. Jos näet haluamasi tiedoston ikkunassa, voit ladata sen myös napsauttamalla sitä kahdesti hiirellä.

Jos haluat aloittaa uuden tiedoston kirjoittamisen, voit aloittaa tyhjästä valitsemalla **File**-valikosta kohdan **New**. Tämä tyhjentää vanhan tiedoston näkyvistä ja voit aloittaa uuden tiedoston kirjoittamisen.

Harjoittele tiedostojen tekemistä, tallentamista ja avaamista. Kokeile tehdä itsellesi kaksi tiedostoa, joita muokkaat vuorotellen.

SIIRTYMINEN OHJELMASTA TOISEEN

Windowsissa on mahdollista pitää useita ohjelmia käynnissä yhtäaikaa. Tällä tavalla voit kirjoittaa tiedostoja editorillasi, käydä välillä selaimessa ja palata takaisin editoriin.

Voit vaihtaa käyttämääsi ohjelmaa kahdella tavalla. Windows 95:ssä näet kaikki käynnissä olevat ohjelmat kuvaruudun alareunan toimintopalkissa.

Voit napsauttaa hiirellä ohjelman kuvaketta, jolloin ohjelman ikkuna tulee päällimmäiseksi ruudulla ja voit käyttää sitä.

Toinen tapa vaihtaa ohjelmaa on pitää pohjassa **Alt**-näppäintä ja painaa yhtäaikaa **Tab**-näppäintä. Tämä vaihtaa taustalla olevan ohjelman päällimmäiseksi ja vie käytössä olleen ohjelman taustalle. Jos pidät **Alt**-näppäintä koko ajan pohjassa, näet listan kaikista käynnissä olevista ohjelmista ja voit siirtyä mihin tahansa ohjelmaan painamalla monta kertaa **Tab**-näppäintä. Kun painat tarpeeksi monta kertaa, pääset takaisin listan alkuun. Tämä tapa toimi myös vanhassa Windowsissa.

Kokeile avata yhtäaikaa **Notepad** ja Netscape ja vaihtaa niiden välillä. Kokeile myös maalata tekstiä Netscapessa näkyvältä sivulta, kopioida se ja liimata se **Notepad**iin.

ENSIMMÄINEN WWW-SIVUSI

TEKSTIPOHJAINEN KOTISIVU

Seuraavaksi siirrytään itse sivujen tekemiseen!

Tässä luvussa opit tekemään itsellesi ensimmäisen kotisivun. Opit myös, miten voit katsella tekemääsi sivua selaimessasi ja miten muokkaat sitä.

MIKÄ ON HTML?

Ensimmäiseksi on aika tutustua kieleen, jolla WWW-sivut on tehty. WWW-sivujen tekeminen on erittäin yksinkertaista. Sivut koostuvat enimmäkseen tavallisesta tekstistä - samasta tekstistä, jota katselet selaimesi ruudulla. Sivujen ulkoasu ja niiden toiminta saadaan aikaan sijoittamalla tekstin sekaan yksinkertaisia HTML-komentoja.

Termi HTML tulee sanoista **Hypertext Markup Language**, eli Hypertekstien merkkauskieli. Jos katselet HTML-kielistä tiedostoa, huomaat että kaikki komennot on merkitty tekstin sekaan suurempi- ja pienempikuin -merkkien sisään. Yksi esimerkki HTML-komennosta on **<p>**, joka aloittaa uuden tekstikappaleen. Alla näet esimerkin siitä, miltä kaksi tekstikappaletta voisivat näyttää tavallisella kotisivulla:

```
<p>Hei, ja tervetuloa kotisivulleni! Toivon että
viihdyt sivuillani ja löydät niiltä paljon
mielenkiintoista luettavaa.
<p>Olen syntynyt Vanuatu-saarilla vuonna 1953. Kun
olin 8 vuoden vanha, muutin vanhempieni kanssa
Suomeen...
```

HTML-komennoilla voi olla aloitus- ja lopetuskomento. Aloituskomento aloittaa jonkin ominaisuuden, vaikkapa tekstin lihavoinnin. Kaikki sen jälkeen tuleva teksti näkyy selaimen ruudulla lihavoituna, kunnes lopetuskomento lopettaa sen. Lopetuskomento näyttää yleensä samanlaiselta kuin aloituskomentokin, mutta se sisältää ylimääräisen kauttaviivan. Esimerkiksi lihavoitu teksti aloitetaan komennolla **** ja lopetetaan komentoon **.** Alla näet esimerkin:

```
Jos käyt joskus Vanuatu-saarilla, <b>muista
ehdottomasti</b> käydä Joen kuppilassa. Sano terveisiä
Annalta.
```

Usein HTML-komento sisältää **attribuutteja**, eli lisäosia jotka tarkentavat komennon toimintaa. Kappaleenvaihtokomentoon ** ** voidaan lisätä attribuutti, joka kertoo että kyseinen kappale täytyy sijoittaa ruudun vasemman reunan sijasta oikeaan reunaan. Silloin komento näyttää tällaiselta:

```
<p align=right>Terveisin, Anna
```

Kaikki attribuutit laitetaan komennon ympärillä olevien suurempi- ja pienempi kuin -merkkien sisään, kuten yllä olevassa esimerkissä. Attribuuttia ei tarvitse toistaa enää lopetuskomennossa, vaan sinne riittää pelkkä komento kauttaviivan kanssa:

```
<h1 align=center>Hei, maailma!</h1>
```

Voit kirjoittaa HTML-komennot isoilla tai pienillä kirjaimilla - selain ei kiinnitä kirjaimien kokoon huomiota. Joskus komennot voivat olla niin pitkiä, että ne täytyy jakaa monelle riville. Tämäkään ei haittaa selailua.

Ja seuraavaksi mennään varsinaisten komentojen pariin!

LEIPÄTEKSTI

Leipäteksti kirjoitetaan HTML-tiedostoon sellaisenaan. Kirjoittamasi teksti tulee näkyviin selaimessasi suhteutetulla tekstityypillä näytettynä, vasen reuna tasattuna ja selaimen ikkunaan sopivan levyisiksi kappaleiksi muotoiltuna.

Selain ei ota huomioon HTML-tiedostoon kirjoittamiasi rivinvaihtoja eikä siellä olevia ylimääräisiä välilyöntejä. Kaikki rivit katsotaan kuuluviksi samaan kappaleeseen, siihen asti kunnes vastaan tulee uuden tekstikappaleen aloittava komento.

Jokaisen tekstikappaleen alkuun tulee komento ** ** ja loppuun lopetuskomento ** **.

Lopetuskomento ei ole pakollinen. Jos jätät tekstikappaleiden lopetuskomennot kirjoittamatta, käyttämäsi selain muotoilee tekstin omalla tavallaan. Selain katsoo tekstikappaleen loppuvan silloin, kun HTML-tiedostossa tulee vastaan uusi kappaletyyli (toinen tekstikappale, otsikko jne.).

Useimmat selaimet osaavat tulkita tekstin leipätekstiksi, vaikka et käyttäisikään aloituskomentoa **<p>** esimerkiksi silloin, kun leipäteksti alkaa kappaleen jälkeen. Mahdollisten väärinkäsitysten välttämiseksi kannattaa kuitenkin käyttää aloituskomentoa jokaisen tekstikappaleen alussa.

Kun selain muotoilee tekstiä ruudulle, se aloittaa kappaleen uudelta riviltä ja jättää kappaleiden väliin hieman tyhjää tilaa.

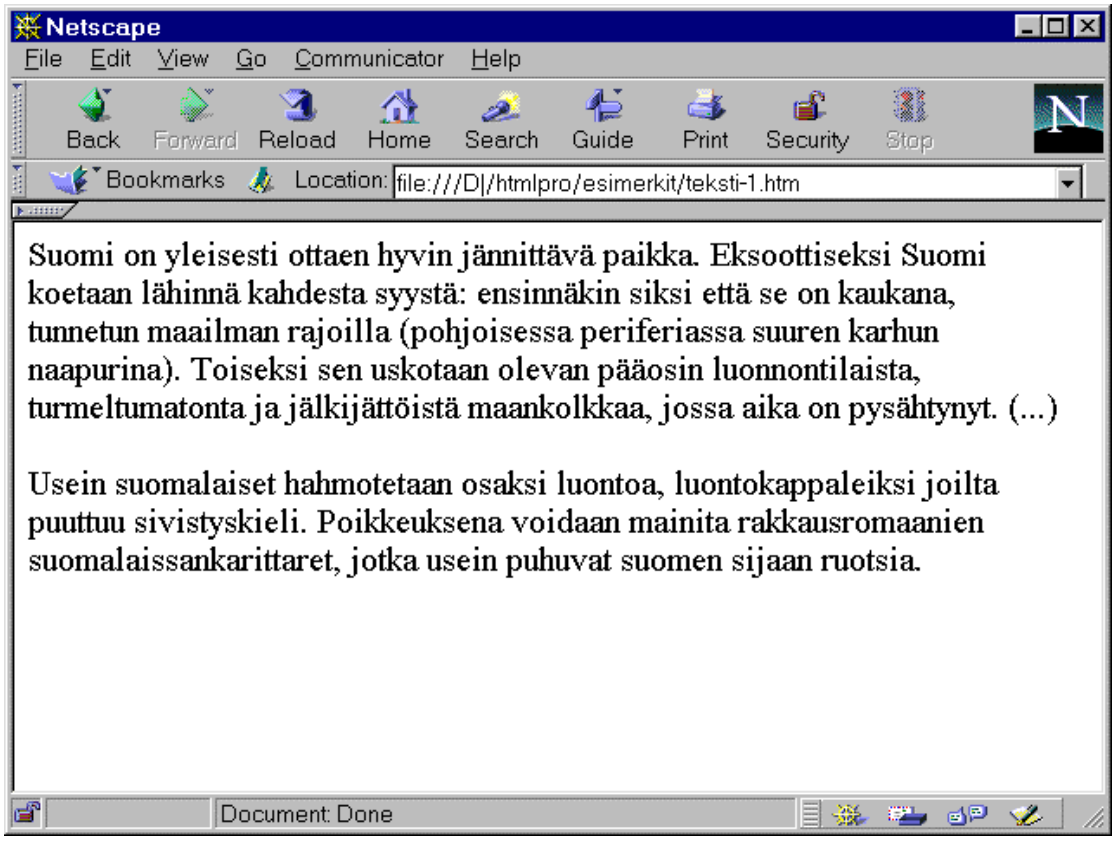
Esimerkki HTML-tiedostossa olevasta leipätekstistä:

```
<p>Suomi on yleisesti ottaen hyvin jännittävä paikka.
Eksoottiseksi Suomi koetaan lähinnä kahdesta syystä:
ensinnäkin siksi että se on kaukana, tunnetun maailman
rajoilla (pohjoisessa periferiassa suuren karhun
naapurina). Toiseksi sen uskotaan olevan pääosin
luonnontilaista, turmeltumatonta ja jälkijättöistä
maankolkkaa, jossa aika on pysähtynyt. (...)</p>


<p>Usein suomalaiset hahmotetaan osaksi luontoa,
luontokappaleiksi joilta puuttuu sivistyskieli.
Poikkeuksena voidaan mainita rakkausromaanien
suomalaissankarittaret, jotka usein puhuvat suomen
sijaan ruotsia.</p>[1]
```

Sama teksti näytölle muotoiltuna tuo näkyviin kaksi tekstikappaletta:

---

[1]Riitta Santala, "Me muut — suomalaiset englanninkielisessä fiktiossa". — *Me ja muut* (toim. Marjo Kylmänen), 1994, ss. 74-75.



Automaattisen tekstinmuotoilun vuoksi sinun kannattaa pitää mielessäsi pari tärkeää seikkaa:

1. Älä tavuta leipätekstiä. Keskeltä tavuviivalla katkaistu sana saattaa tulla lopullisessa tekstissä näkyviin yhdellä rivillä, jolloin tuloksena on hullunkurisen näköistä tekstiä: saa-tat saada ta-vutettuja sanoja keskelle riviä.

2. Älä yritä tehdä tyhjää tilaa sivullesi ylimääräisillä kappalekomennoilla. Useimmat selaimet ohittavat tyhjät kappaleet. Jos haluat välttämättä sijoittaa tekstisi väliin tyhjää tilaa, voit käyttää esimuotoiltua tekstityyppiä. Toinen tapa on tehdä läpinäkyvä kuva, joka on haluamasi tyhjän tilan kokoinen.

3. Älä yritä sisentää tekstiä välilyönneillä tai tabulaattoreilla. Selain heittää kaikki ylimääräiset välilyönnit pois muotoillessaan tekstiä lukijan kuvaruudulle sopivaksi. Sen sijaan voit käyttää välilyöntejä HTML-sivun ulkoasun parantamiseen. Voit lisätä tiedostoosi välilyöntejä ja tehdä siitä itsellesi mahdollisimman selkeän. Koska HTML-tiedostossa olevat sisennykset eivät tule näkyviin lukijalle, vaan näkyvät ainoastaan kirjoittamassasi koodissa, voit käyttää niitä vapaasti omiin tarkoituksiisi.

Jos nimenomaan haluat tehdä sisennettyä tekstiä, voit käyttää siihen tarkoitukseen esimuotoiltua tekstityyppiä. Monilla sivuilla kirjoittajat ovat tehneet sisennettyä tekstiä myös erilaisten listojen avulla — esimerkiksi käyttämällä määritelmälistoja.

Voit keskittää tekstikappaleet kuvaruudun keskelle lisäämällä tekstikappaleen aloituskomentoon attribuutin **`align=center`**. Esimerkiksi:

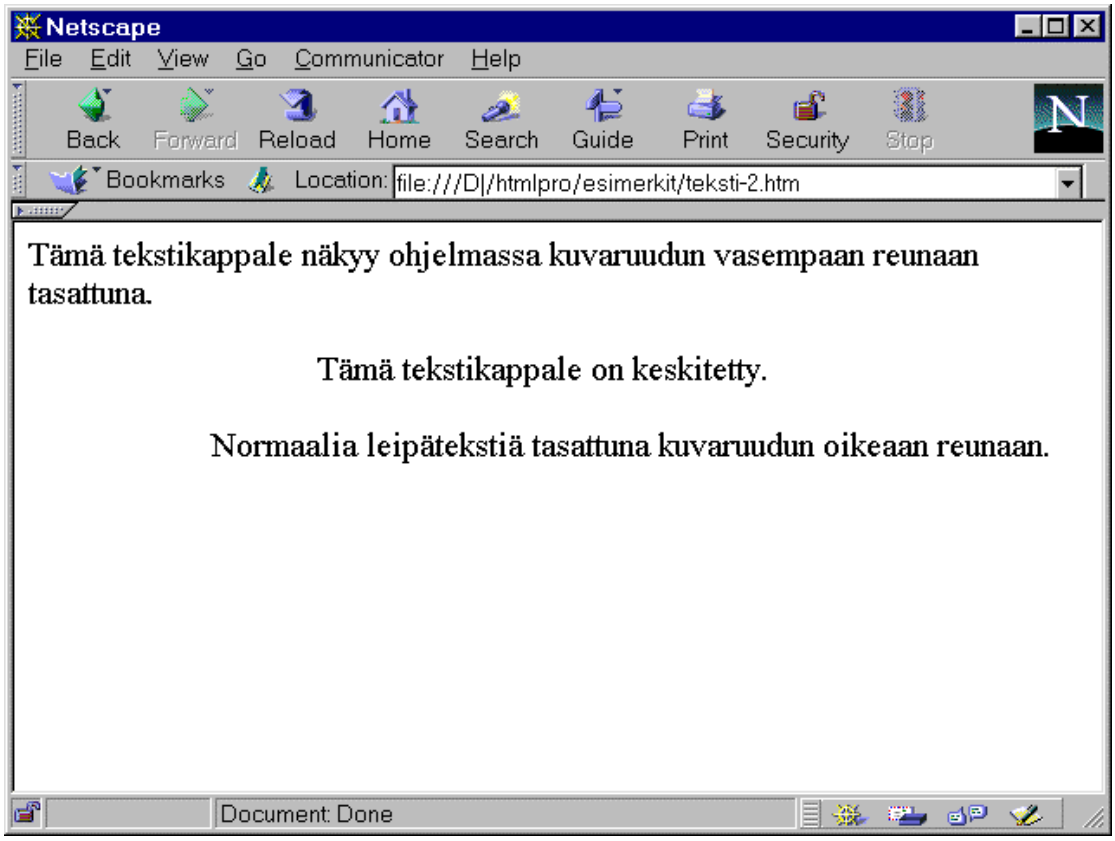
```
<p>Tämä tekstikappale näkyy ohjelmassa kuvaruudun
vasempaan reunaan tasattuna.</p>
<p align=center>Tämä tekstikappale on keskitetty.</p>
```

Tämä komento keskittää myös kappaleessa olevat kuvat.

Jos haluat tasata leipätekstin oikean reunan vasemman sijasta, voit käyttää kappalekomennossa attribuuttia **`align=right`**.

```
<p align=right>Normaalia leipätekstiä tasattuna
kuvaruudun oikeaan reunaan.
```

Selaimen ruudulla esimerkit näyttävät seuraavanlaisilta:



Jos haluat sijoittaa tekstiin rivinvaihdon, joka ei kuitenkaan vaihda kappaletta, voit käyttää rivinvaihtokomentoa **
**. Komennon jälkeen tuleva teksti alkaa uudelta riviltä, mutta sen ja edellisen rivin väliin ei tule ylimääräistä tilaa. Rivinvaihtokomentoa saatat tarvita esimerkiksi osoitteita kirjoittaessasi:

```
<p>Aku Ankka<br>
Paratiisitie 13<br>
13131 Ankkalinna</p>
```

OTSIKOT

Otsikko on tavallista leipätekstiä suuremmalla kirjasimella näkyvää tekstiä, joka on erotettu ympäröivästä tekstistä tyhjillä riveillä. Otsikkoja on monenkokoisia: sivun pääotsikko näytetään isoilla kirjaimilla, kun taas lukujen väliotsikot koostuvat hieman pienemmistä kirjasintyypeistä.

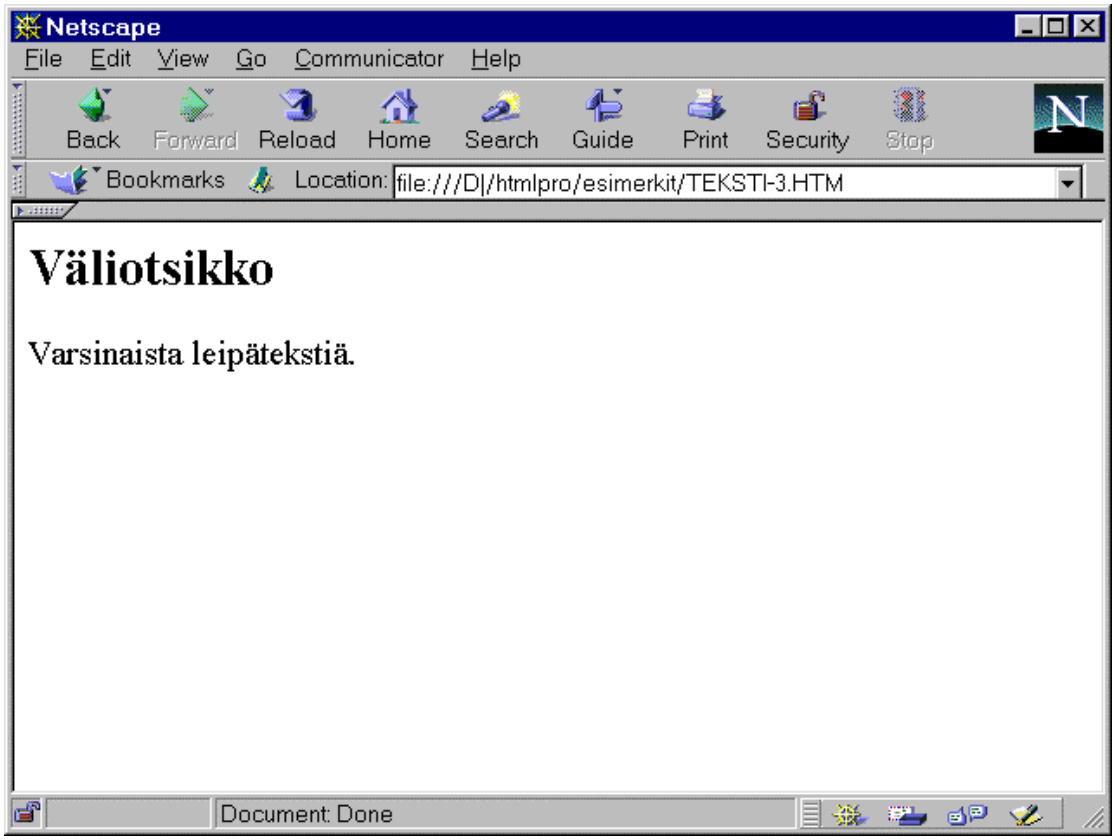
WWW-sivuilla voi olla kuudenkokoisia otsikkoja. Suurinta ja tärkeintä otsikkoa, pääotsikkoa, merkitään tasolla yksi. Taso kaksi on hieman pienempi otsikko, taso kolme jälleen pienempi, ja niin edelleen. Otsikko numero kuusi on pienin väliotsikon taso.

Otsikko merkitään tekstiin komennolla **<h*x*>**, jossa *x* on otsikon tasonumero. Otsikon tekstin jälkeen merkitään lopetuskomento, joka on **</h*x*>**. Tason yksi otsikko merkittäisiin siis komentojen **<h1>** ja **</h1>** väliin.

Kaikki aloitus- ja lopetuskomennon väliin jäävä teksti kuuluu otsikkoon. Itse otsikko tulostuu omalle rivilleen, pienen välimatkan päähän varsinaisesta leipätekstistä:

```
<h2>Väliotsikko</h2>
<p>Varsinaista leipätekstiä.</p>
```

Sama sivu muotoiltuna kuvaruudulle näyttää seuraavanlaiselta:



Kun otsikko muotoillaan kuvaruudulle, sen ylä- ja alapuolelle jätetään aina jonkin verran tyhjää tilaa. Kaikki otsikon jälkeen tuleva teksti alkaa uudelta riviltä.

Jos haluat keskittää otsikon sivun keskelle vasemman reunan sijasta, voit lisätä otsikkokomentoon attribuutin **`align=center`**.

Saat sijoitettua otsikon kuvaruudun oikeaan laitaan lisäämällä otsikkokomentoon attribuutin **align=right**. Esimerkki:

```
<h1 align=right>Uusinta uutta</h1>
```

Kuten kaikki muutkin HTML:n tekstityypit, myös otsikot ovat loogisia kokonaisuuksia. Se tarkoittaa, että et voi kirjoittaessasi tietää lukijan kuvaruudulla näkyvän otsikon kokoa. Ainoa, mihin voit luottaa ovat otsikoiden väliset suhteet — tason 1 otsikko on suurempi kuin tason 2 otsikko, ja niin edelleen.

Voit käyttää otsikoita muutenkin kuin numerojärjestyksessä: HTML ei estä käyttämästä pelkästään esimerkiksi tason 1 ja tason 6 otsikoita ilman väliin jääviä tasoja. Se saattaa tosin näyttää joidenkin lukijoiden näytöllä erikoiselta eikä se ole sivun loogisen rakenteen kannalta suositeltavaa.

KATSELU SELAIMESSA

Nyt, kun olet oppinut HTML:n peruskomennot, on aika tehdä ensimmäinen kotisivu ja kokeilla sen katselemista selaimessa.

Ota esiin editorisi ja kirjoita siihen yksinkertainen peruskotisivu käyttäen aikaisemmissa luvuissa oppimiasi komentoja. Alla näet minun esimerkkini - voit tietysti kirjoittaa omalle kotisivulle haluamiasi tekstejä!

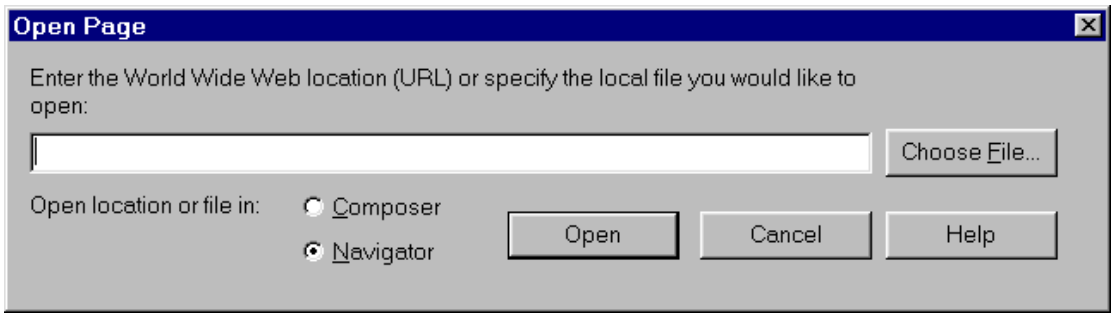
```
<h1>Samin kotisivu</h1>
<p>Tämä on ensimmäinen kotisivuni! Odotan
jännityksellä sitä, miltä se näyttää selaimen
ruudulla.</p>
<p align=right>Tekijä: Sami, 1997</p>
```

Kun olet kirjoittanut kotisivusi, tallenna se tietokoneesi kovalevylle HTML-tiedostoja varten tekemääsi hakemistoon. Tallenna sivu nimellä **kotisivu.htm**. Jos käytät HTML-editoria, se tekee sinulle automaattisesti tiedostonnimen tarkentimen **.htm**. Jos käytät muokkaamiseen jotain tavallista tekstieditoria, kuten Windowsin **Notepad**ia, älä unohda varmistaa, että tiedoston päätteeksi ei tule jotain muuta.

Kun kotisivusi on tallentunut, voit katsella sitä ensimmäisen kerran. Jätä editorisi käyntiin (voit kutistaa sen napsauttamalla ikkunan ylänurkassa

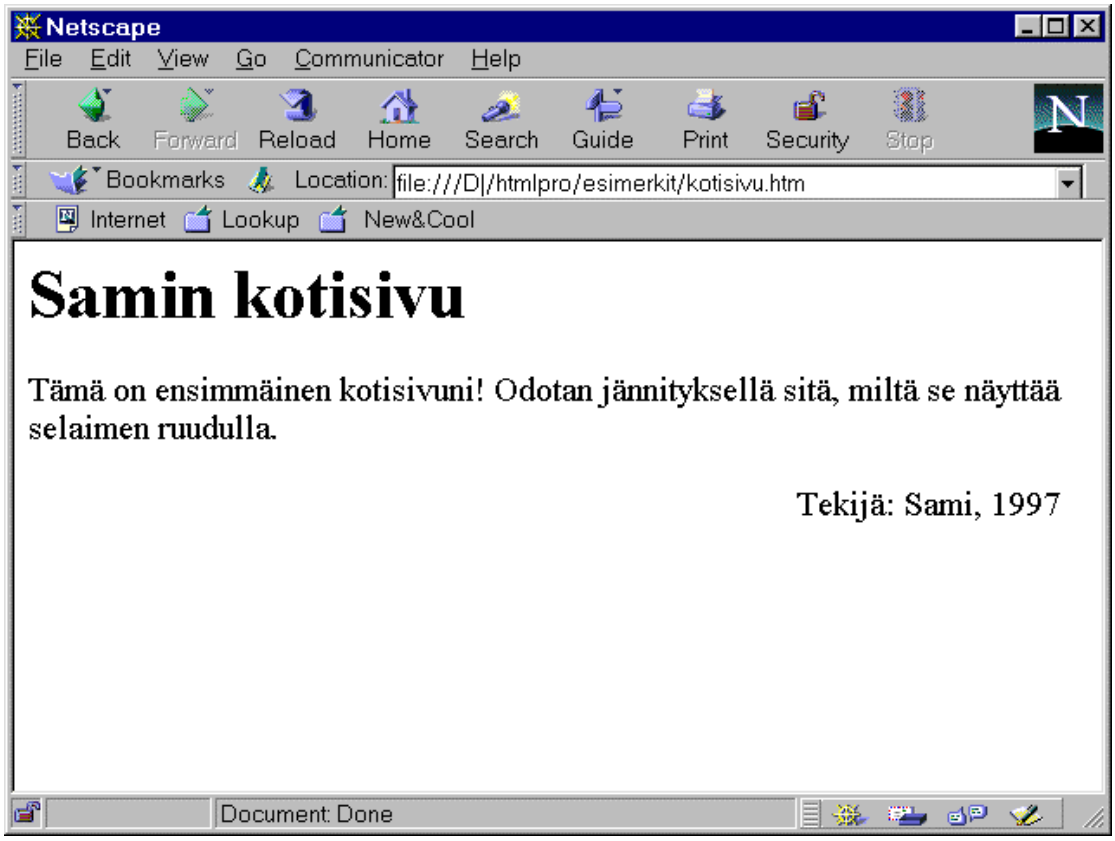
olevaa kutistuspainiketta). Käynnistä sen jälkeen selaimesi aivan normaalisti.

Kovalevylle tallennetut HTML-tiedostot ladataan selaimeen yleensä **Open File** -nimisellä toiminnolla tai vastaavalla. Netscapessa pääset lataamaan tiedoston kovalevyltä, kun otat ensin **File**-valikosta toiminnon **Open**, ja esille tulevasta ikkunasta toiminnon **Choose File**. Internet Explorerissa saman toiminnon nimi on **Browse** (niin ikään **Open File** -ikkunassa).



Kun napsautat **Choose File** -painiketta, saat näytöllesi samanlaisen tiedostonavausikkunan kuin muissakin ohjelmissa. Etsi kovalevyltä äsken tallentamasi tiedosto ja avaa se.

Juuri tekemäsi kotisivu tulee näkyviin selaimesi ruudulle - onneksi olkoon, olet tehnyt juuri elämäsi ensimmäisen kotisivun!



Kun haluat tehdä muutoksia tekemääsi sivuun, tuo takaisin esille HTML-editorisi (Windowsissa voit vaihtaa ohjelmaa painamalla yhtäaikaa **Alt**- ja **Tab**-näppäimiä kunnes näet editorin). Tee pari muutosta sivuun - lisää vaikka väliin uusi tekstikappale tai muuta otsikkoa. Kun olet tehnyt muutokset, tallenna tekemäsi muutokset ja siirry takaisin selaimeen. Kun painat selaimen **Reload**-painiketta (Internet Explorerissa **Refresh**-painiketta), näet kuvaruudullasi juuri tekemäsi muutokset.

Tällä tavalla voit muokata sivuasi ja tarkastella tekemiäsi muutoksia selaimesi näytöllä. Aina kun olet tehnyt muutoksia, muista ensin tallentaa sivusi. Tämän jälkeen voit siirtyä takaisin selaimeen ja painaa **Reload**-painiketta, jolloin näet tekemäsi muutokset selaimessasi.

Nyt on sopiva hetki kokeilla aiemmin näytettyjä komentoja ja tutkia, miltä ne näyttävät selaimesi ruudulla. Tällä tavalla tulet tutuksi oppimiesi komentojen kanssa.

Kun olet saanut kotisivustasi haluamasi näköisen, tee seuraavaksi toinen HTML-sivu. Aloita editorissa uuden tiedoston editointi (useimmissa editoreissa tämä tapahtuu valitsemalla **File**-valikosta toiminto **New**). Tee tästä sivusta vaikka sivu, jossa kerrot yhteystietosi:

```
<h1>Yhteystiedot</h1>
<p>Voit saada minuun yhteyden monella eri tavalla:
<h2>Puhelin</h2>
<p>Puhelinnumero: (03) 555 1234
<br>Kännykkä: 040-555 9876
<h2>Internet</h2>
<p>Sähköposti: sami@provider.fi
<br>Kotisivu: http://www.provider.fi/~sami
```

Huomaa rivinvaihtokomennon käyttö: **
**-komennollahan sait tehtyä pelkän rivinvaihdon, joka ei tee rivien väliin tyhjää tilaa kuten kappaleenvaihtomerkki.

Tallenna tämä sivu kovalevyllesi uudella nimellä - nimeksi voit antaa vaikkapa **osoite.htm**. Siirry seuraavaksi selaimeesi, avaa sivu kovalevyltä ja tarkista että siitä tuli haluamasi näköinen. Jos haluat tehdä muutoksia, voit tehdä ne samalla tavalla kuin kotisivusi kanssa. Myöhemmin opit, kuinka voit liittää nämä kaksi sivua toisiinsa linkeillä.

VIRHEITÄ KOMENNOISSA?

Virheiden syntymistä ei voi välttää - tämä pitää paikkansa myös WWW-sivuja tehtäessä. Toisin kuin ohjelmoitaessa, WWW-sivuja tehtäessä virheet eivät estä sivun katselemista. Jos kirjoitat vahingossa virheellisen HTML-komennon, selain ei anna siitä minkäänlaista virheilmoitusta vaan jättää kyseisen komennon kokonaan huomiotta.

Virheellisten komentojen huomiotta jättämisessä on sekä hyviä että huonoja puolia. Huono puoli on se, että virheellinen komento saattaa jäädä kokonaan huomaamatta tai sen etsimiseen kuluu paljon aikaa. Otetaan esimerkkinä HTML-sivu, joka sisältää otsikon ja kappaleen leipätekstiä. Sivulla on kuitenkin virhe: otsikon lopetuskomentoon on tullut kirjoitusvirhe.

```
<h1>Otsikko</j1>
<p>Tekstikappale
```

Kun selain näyttää sivua ruudulla, se kohtaa ensimmäisenä otsikon aloituskomennon. Se ei kuitenkaan löydä koskaan lopetuskomentoa (komento **/j1** on selaimelle tuntematon komento, joten se ei välitä siitä mitään).

Siksi lukija näkee selaimessaan sivun, jonka teksti on alusta loppuun asti yhtä isoa otsikkoa. Muita yleisiä kirjoitusvirheitä HTML-komennossa ovat:

- Lopetuskomennosta puuttuu kauttaviiva
- Komennon kulmasulut ovat väärinpäin (esimerkiksi **<h1<**)
- Attribuuteissa on vain yksi lainausmerkki
- Attribuutteja ei ole erotettu välilyönnillä

Virheellisten komentojen ohittamisesta on myös hyötyä. Vanhemmat selaimet eivät välttämättä osaa kaikkia uusien selaimien ymmärtämiä HTML-komentoja. Se ei kuitenkaan estä sivujen lukemista - vaikka sivulla on käytetty uusia komentoja, vanhemmat selaimet osaavat silti näyttää sivun enemmän tai vähemmän oikein.

Jos sivulla on käytetty vaikkapa tekstin ulkoasuun vaikuttavaa **font**-komentoa ja sitä luetaan vanhemmalla selaimella, joka ei ymmärrä tällaista komentoa, teksti on silti luettavissa. Tekstityypit vain jäävät näkymättä.

LISÄÄ HTML-KOMENTOJA

Seuraavaksi paneudutaan vähän syvemmälle HTML:ään ja katsotaan komentoja, joilla voit lisätä tekstiisi eloa. Voit kokeilla seuraavassa esitettyjä komentoja oman kotisivusi kanssa. Kokeile annettuja komentoja, lisäile niitä sivuillesi ja katso, miten ne vaikuttavat omaan kotisivuusi.

TEKSTIMUOTOILU

Tekstimuotoiluilla voit vaikuttaa yksittäisten merkkien tai sanojen muotoiluun. Nämä sisältävät erilaisia tekstin korostuksia (esimerkiksi lihavointi ja kursiivi) ja loogisia tekstityyppejä (esimerkiksi korostettu teksti ja vahvasti korostettu teksti).

Loogisen ja fyysisen tekstimuotoilun erot ovat muotoillun tekstin toteutustavassa. Jos merkitset tekstin kursiiviksi, se tulostuu kursiivina niillä näytöillä, jotka pystyvät sellaista esittämään. Muilla se yleensä jätetään tulkitsematta ja esitetään sen sijaan tavallisena tekstinä. Jos taas merkitset tekstin korostetuksi, se saatetaan tulostaa kursiivilla, lihavoidulla tekstillä tai vaikkapa isommalla kirjasinkoolla, riippuen siitä mihin lukijan

näyttö pystyy ja miten selaimessa on määritelty tämän tekstityypin esitystapa.

Kun kirjoitat HTML-sivua, on suositeltavaa käyttää loogisia tekstitehosteita fyysisten muotoilujen sijasta, ellet nimenomaan halua käyttää jotain tiettyä tyyliä. Selaimien ja näyttölaitteiden kehittyessä loogisille tekstimäärittelyille voidaan kehitellä uusia efektejä — tulevaisuuden selain saattaa esimerkiksi näyttää korostetun tekstin eri värillä tai marginaaliin merkityllä erikoistehosteella.

FYYSISET MUOTOILUKOMENNOT

Fyysiset muotoilukomennot kertovat selaimelle, millaisella tekstityylillä merkitty teksti tulisi näyttää ruudulla. Lopullinen esitystapa on selaimen päätettävissä — jos tekstityyli on käytettävissä, ohjelma käyttää sitä. Muussa tapauksessa selain voi jättää muotoilutyylin kokonaan esittämättä tai käyttää pyydetyn tyylin sijasta jotain muuta tyyliä.

Komennolla **** aloitetaan lihavoitu teksti. Lihavointi jatkuu aina päättymiskomentoon **** saakka.

Kursiiviteksti merkitään komennoilla **** ja ****. Jos mahdollista, se näytetään lukijan näytöllä aitona kursiivileikkauksena. Jos lukijan kone ei pysty tähän, selain käyttää sen sijasta kallistettua tekstiä. Silloin kun kursiivia ei ole mahdollista näyttää (esimerkiksi tekstipäätteillä), selain saattaa esittää tekstin mahdollisuuksien mukaan muulla tavalla muotoiltuna.

Alleviivaus aloitetaan **<u>**-komennolla ja lopetetaan komentoon **</u>**. Monet nykyisistä selaimista eivät osaa vielä toteuttaa alleviivattua tekstiä. Tällöin selaimet jättävät alleviivauksen kokonaan näyttämättä, tai käyttävät sen tilalla jotain muuta tekstityyppiä.

Kirjoituskonetyyli koostuu tasalevyisistä merkeistä. Yleensä tämä tyyli esitetään selaimessa `Courier-kirjasintyylillä` tai jollain muulla tasalevyisellä kirjasintyylillä. Kirjoituskonetyyli merkitään komennolla **<tt>**. Jos selain ei osaa näyttää tyyliä, se jättää sen huomiotta tai korvaa sen jollain muulla.

Kun haluat tehdä tekstiä, jonka kirjasintyyli on normaalia leipätekstiä pienempää, voit käyttää **<small>**-komentoa. Selain esittää määritellyn kohdan "pienellä painettuna", jos se on mahdollista.

Vastaavasti voit tehdä normaalia leipätekstiä suurempaa tekstiä komennolla **<big>**.

Komennolla **<sup>** voit tehdä yläindeksejä esimerkiksi matemaattisiin kaavoihin tai viitteiden esittämiseen. Yläindeksit näkyvät ruudulla mahdollisuuksien mukaan pienemmällä tekstityypillä ja normaalia leipätekstiä ylempänä.

Alaindeksi on yläindeksin "vastakohta": **<sub>** -komennolla merkityt tekstit näkyvät normaalia tekstiä alempana ja pienemmällä tekstityypillä.

Tekstin päälle vedetty yhtenäinen viiva saadaan aikaan **<strike>** -komennolla.

Voit tehdä yhdistettyjä tyylejä, esimerkiksi ***lihavoitua kursiivia*** sijoittamalla tyylikomennot sisäkkäin:

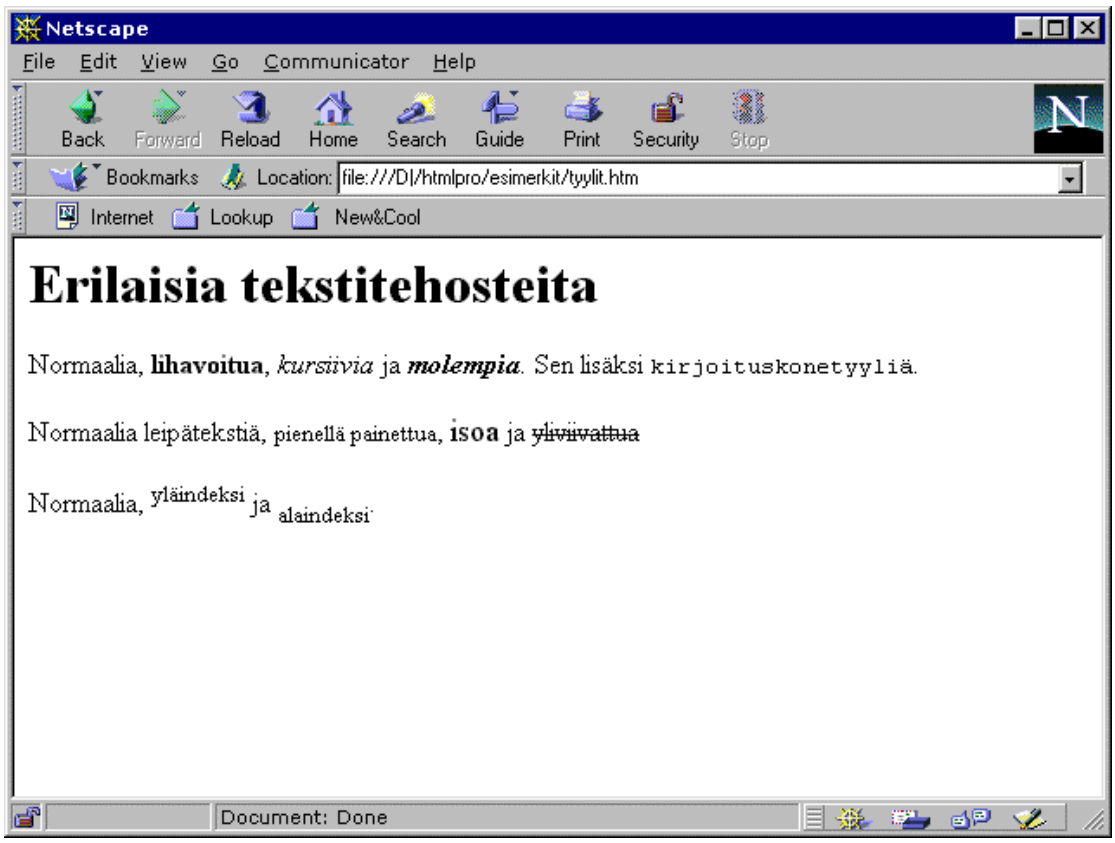
```
<b><i>lihavoitua kursiivia</i></b>
```

Muista kuitenkin sijoittaa tekstiin lopetuskomennot oikeassa järjestyksessä: yllä olevassa esimerkissä ovat esimerkiksi kursiivikomennot, joiden ympärillä on lihavointikomennot.

Alla olevassa esimerkissä on käytetty aiemmin esiteltyjä tyylejä:

```
<h1>Erilaisia tekstitehosteita</h1>

<p>Normaalia, <b>lihavoitua</b>, <i>kursiivia</i> ja
<b><i>molempia</i></b>. Sen lisäksi
<tt>kirjoituskonetyyliä</tt>.

<p>Normaalia leipätekstiä, <small>pienellä
painettua</small>, <big>isoa</big> ja
<strike>yliviivattua</strike>

<p>Normaalia, <sup>yläindeksi</sup> ja
<sub>alaindeksi</sub>.
```

Selaimessa tyylit näyttävät tältä:



LOOGISET MUOTOILUKOMENNOT

Selaimissa on myös käytössä useita erilaisia loogisia muotoiluja erityyppisiä tekstejä varten. Loogiset muotoilut eivät kerro sitä, miten teksti tulee esittää ruudulla, vaan ne kertovat minkä tyyppistä merkitty teksti on. Selain itse päättää sille annettujen ohjeiden perusteella, miten tietty looginen tyyli esitetään näytöllä.

Loogisten tyylien avulla erityyppiset tietokoneet tai laitteet voivat esittää erilaiset korostukset parhaalla mahdollisella tavalla. Klassinen esimerkki on WWW-sivua ääneen lukeva puhesyntetisaattori, joka osaa painottaa sopivalla äänensävyllä korostettavaksi merkittyä tekstiä.

Loogisia tyylejä on kahdeksan kappaletta. Sitaateilla (**Citation**) merkitään erisnimiä, lainauksia jne. Tällä tyylillä voit esittää esimerkiksi kirjan tai kirjailijan nimen:

```
John Dalmas: <cite>The Regiment's War</cite>.
```

Koodi-tyyliä (**Code**) voit taas käyttää, kun haluat liittää mukaan pätkän tietokoneohjelmaa tai muuta koodia. Esimerkiksi:

```
Voit tulostaa muuttujan arvon antamalla komennon
<code>print $1</code>
```

Koodi tulostetaan ruudulle tasalevyisellä fontilla, jos mahdollista. Tämä tyyli on tarkoitettu lyhyiden sanojen tai lauseiden esittämiseen — pidempiin pätkiin kannattaa käyttää **pre**-tyyliä (tästä lisää myöhemmin).

Määrittely (**Definition**) on tyyli erilaisten termien kuvaamiseen. Esimerkiksi:

```
<dfn>Selain</dfn> on ohjelma, jolla voit lukea WWW-
sivuja.
```

Korostus (**Emphasis**) on tarkoitettu erilaisten tärkeiden sanojen tai lauseiden korostamiseen. Esimerkki:

```
Muista, että <em>levykettä ei saa poistaa levyasemasta
valon palaessa!</em>.
```

Näppäily-tyyliä (**Keyboard**) käytetään, kun sivun lukijalle annetaan esimerkki näppäiltäväksi tarkoitetusta tekstistä. Esimerkiksi:

```
Unix-työasemassa Mosaic-selain käynnistetään
komennolla <kbd>xmosaic</kbd>.
```

Esimerkki-tyyliä (**Sample**) voi käyttää yhdessä **<kbd>** –komennon kanssa. Sillä merkitään vaikkapa tietokoneen kuvaruudulle tulostamat tekstit.

```
Jos saat virheilmoituksen <samp>Command not
found</samp>, koneessasi ei ole Mosaic-ohjelmaa.
```

Vahvaa korostusta (**Strong Emphasis**) voit käyttää erityisen tärkeiden asioiden korostamiseen. Sen tarkoituksena on tulostua kuvaruudulla astetta voimakkaampana kuin tavallinen korostus (****). Esimerkki:

```
<strong>Älä koskaan kosketa sähkölaitetta, kun olet
märkä!</strong>
```

Muuttuja-tyylillä (**Variable**) merkitään erilaisia muuttujia, joita käytetään tietokoneohjelmissa tai ohjelmien käynnistämisen yhteydessä. Käytä tätä yhdessä vaikkapa **<kbd>** –tyylin kanssa:

```
Kirjoita <kbd>cd</kbd> <var>hakemisto</var>, kun
haluat siirtyä toiseen hakemistoon.
```

VÄLIVIIVAT

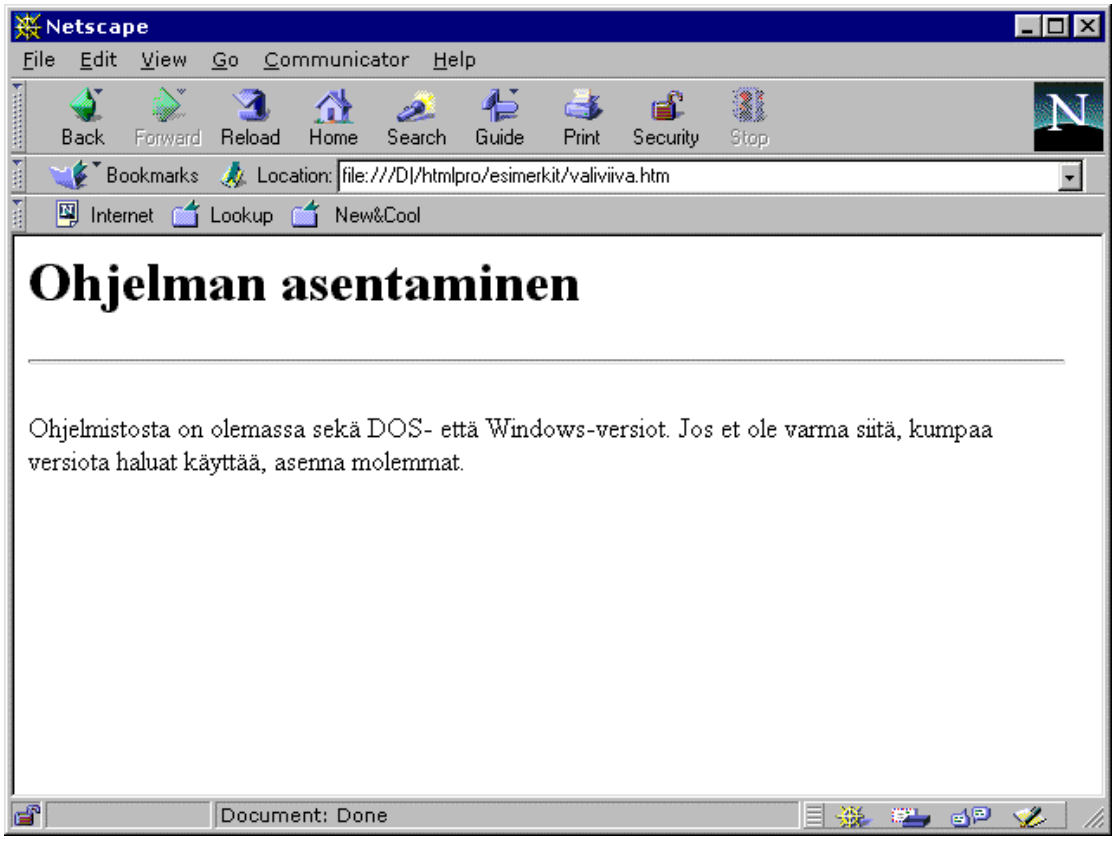
Jos haluat jakaa ruudullasi näkyvän tekstin selkeiksi osiksi, voit sijoittaa kahden kappaleen väliin kuvaruudun poikki ylettyvän viivan komennolla **<hr>**. Väliviiva vaikuttaa tekstimuotoiluun samalla tavalla kuin kappaleenvaihtokin: sen jälkeen tuleva teksti alkaa uudelta riviltä. Väliviiva erottuu edellisestä ja seuraavasta kappaleesta pienellä tyhjällä välillä ja se ulottuu kuvaruudulla vasemmasta oikeaan marginaaliin.

Väliviivaa voi käyttää koristeena tai sivun rakenteen selventäjänä. Sillä voit esimerkiksi erottaa sivun leipätekstin allekirjoituksesta tai ingressin leipätekstistä.

Esimerkki väliviivan käytöstä:

```
<h1>Ohjelman asentaminen</h1>
<hr>
<p>Ohjelmistosta on olemassa sekä DOS- että Windows-
versiot. Jos et ole varma siitä, kumpaa versiota
haluat käyttää, asenna molemmat.</p>
```

Valmiiksi muotoiltu lopputulos:



Normaalisti väliviiva on kuvaruudun reunasta reunaan ulottuva ohut viiva. Nykyaikaisissa selaimissa on mahdollista muuttaa viivan ulkonäköä kuvaruudulla. Niissä ohjelmissa, jotka eivät ymmärrä näitä määrittelyitä, kaikki viivat näkyvät samankokoisina.

Viivan paksuutta voi vaihtaa attribuutilla **size**. Normaalia paksumman viivan voit luoda siis komennolla

```
<hr size=5>
```

Viivan leveyttä ja sen sijaintia ruudulla voi säätää attribuuteilla **width** ja **align**. Esimerkiksi väliviivan, jonka leveys on puolet selaimen ikkunan leveydestä, ja joka on sijoitettuna sivun oikeaan reunaan, voi luoda komennolla

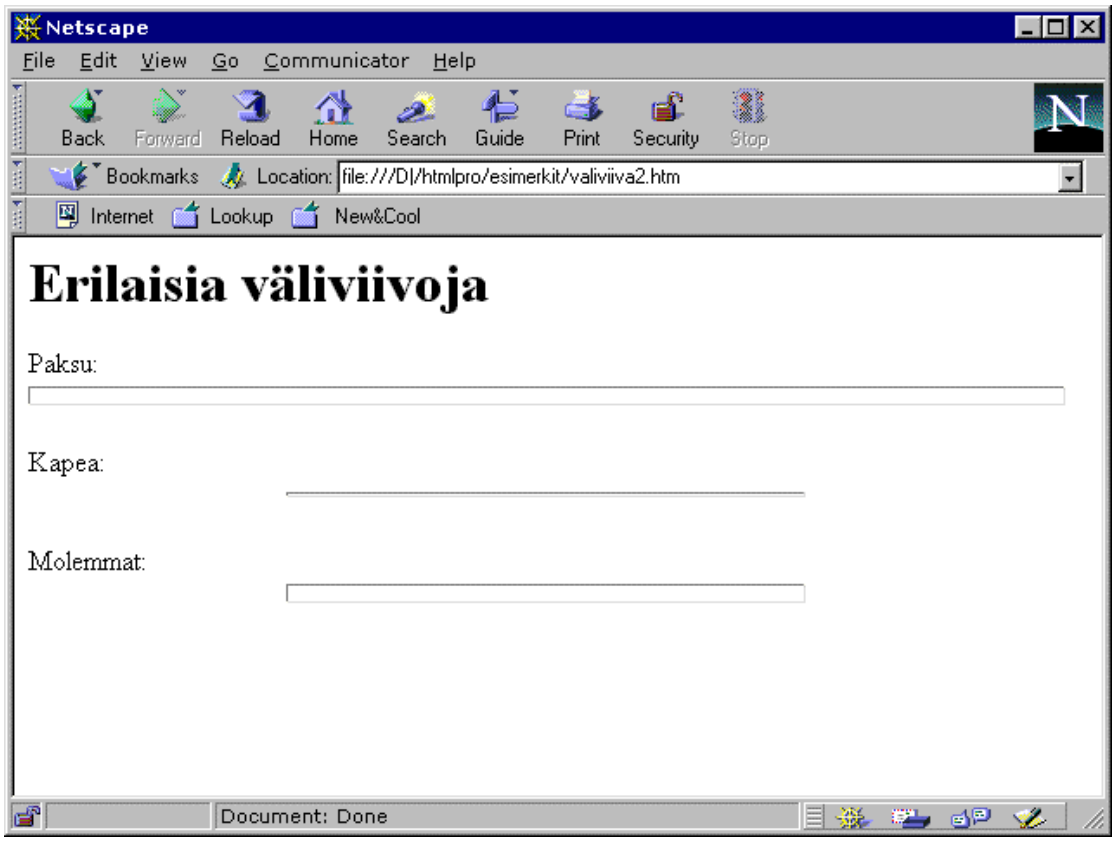
```
<hr width=50% align=right>
```

Jos haluat, että väliviiva ei ole varjostettu vaan pelkkä yksinkertainen viiva, voit lisätä **hr**-komentoon attribuutin **noshade**.

Esimerkki erilaisista väliviivoista:

```
<h1>Erilaisia väliviivoja</h1>
<p>Paksu:
<hr size=10>
<p>Kapea:
<hr width="50%">
<p>Molemmat:
<hr size=10 width="50%">
```

Sama selaimen ruudulla:



ÄÄKKÖSET JA ERIKOISMERKIT

Ääkköset tarkoittavat suomalaisia kirjaimia Å, Ä ja Ö. Jos kirjoitat HTML-tiedostoja Windows-koneilla, sinulla ei todennäköisesti tule ongelmia ääkköskirjaimien kanssa, sillä Windowsissa käytetään samaa merkistöä kuin WWW-sivuilla on sovittu käytettäväksi.

Joissain koneissa ääkköset eivät kuitenkaan toimi aivan samalla tavalla. Jos olet kokeillut kirjoittaa kotisivuja Macintoshilla, olet ehkä törmännyt

siihen, että editorissa kirjoittamasi merkit eivät näytäkään enää samoilta selaimen kuvaruudulla.

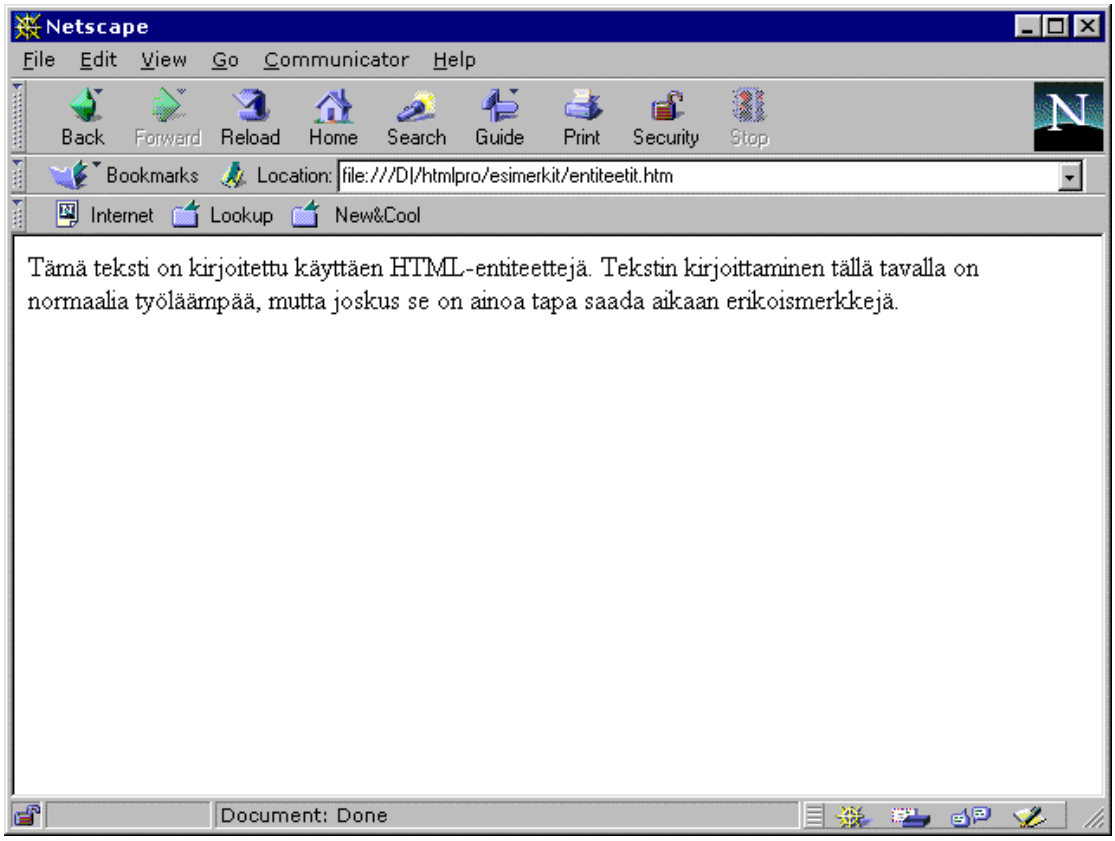
WWW tarjoaa mahdollisuuden kirjoittaa sivuille myös sellaisia merkkejä, joita näppäimistöltä ei saa normaalisti kirjoitettua. Tämä tapahtuu käyttämällä HTML:n *entiteettejä*, eli komentoja jotka luovat selaimen ruudulle merkin.

Jos et esimerkiksi pysty kirjoittamaan näppäimistöltäsi ä-kirjainta, voit käyttää sen sijasta HTML-tiedostossa komentoa **ä**. Ison Ä-kirjaimen saat entiteetillä **Ä**.

Esimerkki HTML-tiedostosta, jossa on käytetty entiteettejä:

```
<p>T&auml;m&auml; teksti on kirjoitettu
k&auml;ytt&auml;en HTML-entiteettej&auml;. Tekstin
kirjoittaminen t&auml;ll&auml; tavalla on normaalia
ty&ouml;l&auml;&auml;mp&auml;&auml;, mutta joskus se
on ainoa tapa saada aikaan erikoismerkkej&auml;.
```

Selaimen ruudulla katsellessa merkkikomennot on muutettu takaisin normaaleiksi kirjaimiksi:



Monet HTML-editorit antavat kirjoittaa tekstiä normaalisti näppäimistöltä, mutta muuttavat tallennusvaiheessa kirjoittamasi ääkköset entiteettimuotoon. Entiteettien käsin kirjoittaminen on harvinaisempaa.

Koska merkit **<** ja **>** on yleensä varattu HTML-komentojen merkitsemiseen, et voi aina käyttää niitä sellaisenaan HTML-tiedostossasi. Jos haluat kirjoittaa sivullesi merkin **<**, voit käyttää sen sijasta entiteettiä **`<`**. Merkin **>** saat aikaan entiteetillä **`>`**. Koska kaikki entiteetit alkavat et-merkillä, myös sille on tehty oma entiteetti: **`&`**.

Entiteetti joka ei näytä kuvaruudulla miltään, mutta on silti hyödyllinen on **` `**. Sen avulla saat tehtyä välilyöntimerkin, joka pitää sanat yhdessä. Kun haluat pitää kaksi sanaa aina yhdessä, voit sijoittaa niiden väliin tämän komennon, esimerkiksi:

```
File Open -komento
```

Vaikka tuossa olevat sanat "**File Open** -komento" sijoittuisivat näytöllä rivin loppuun, niitä ei sijoiteta eri riveille vaan sanat siirtyvät seuraavalle riville yhdessä.

Myös monet ulkomaiset merkit, joita ei löydy suomalaisista näppäimistöistä, on mahdollista toteuttaa käyttäen entiteettejä. Täydellisen listan merkeistä ja entiteeteistä, joilla ne saa näkyviin, löydät liitteestä.

KUVITETTU KOTISIVU

Suurin osa WWW-sivuista sisältää yhden tai useampia kuvia. Koska kuvien lisääminen omille sivuille on erityisen helppoa WWW-sivuja tehtäessä, tämä ei olekaan mikään ihme. Tässä luvussa opit tekemään omia kuvia ja lisäämään niitä sivuillesi.

Kotisivua tehtäessä kuvat valmistetaan eri paikassa kuin sivusi komennot ja teksti. Tämän vuoksi joudut nyt unohtamaan hetkiseksi HTML-editorisi ja siirtymään kokonaan uuteen ohjelmaan. Voit kutistaa HTML-editorisi taustalle odottamaan tai sammuttaa sen kokonaan - tarvitset sitä seuraavan kerran vasta sitten, kun ensimmäinen kuvasi on valmis.

KUVANKÄSITTELYOHJELMA

Kuvien tekemiseen tarvitset jonkinlaisen kuvankäsittelyohjelman. Ohjelmia on monia erilaisia maksuttomista julkisohjelmista aina kalliisiin kuvankäsittelyohjelmiin. Voit käyttää kuviesi tekemiseen miltei mitä tahansa ohjelmaa - ainoana vaatimuksena on se, että käyttämäsi ohjelma osaa tallentaa kuvia **gif**- tai **jpeg**-muodossa (tästä lisää myöhemmin).

Yksi kotisivujen tekijöiden keskuudessa suosituimpia kuvankäsittelyohjelmia on ohjelma nimeltään **Paint Shop Pro**. Se on julkisohjelma, joten voit ottaa sen käyttöön kokeiluajaksi maksutta.

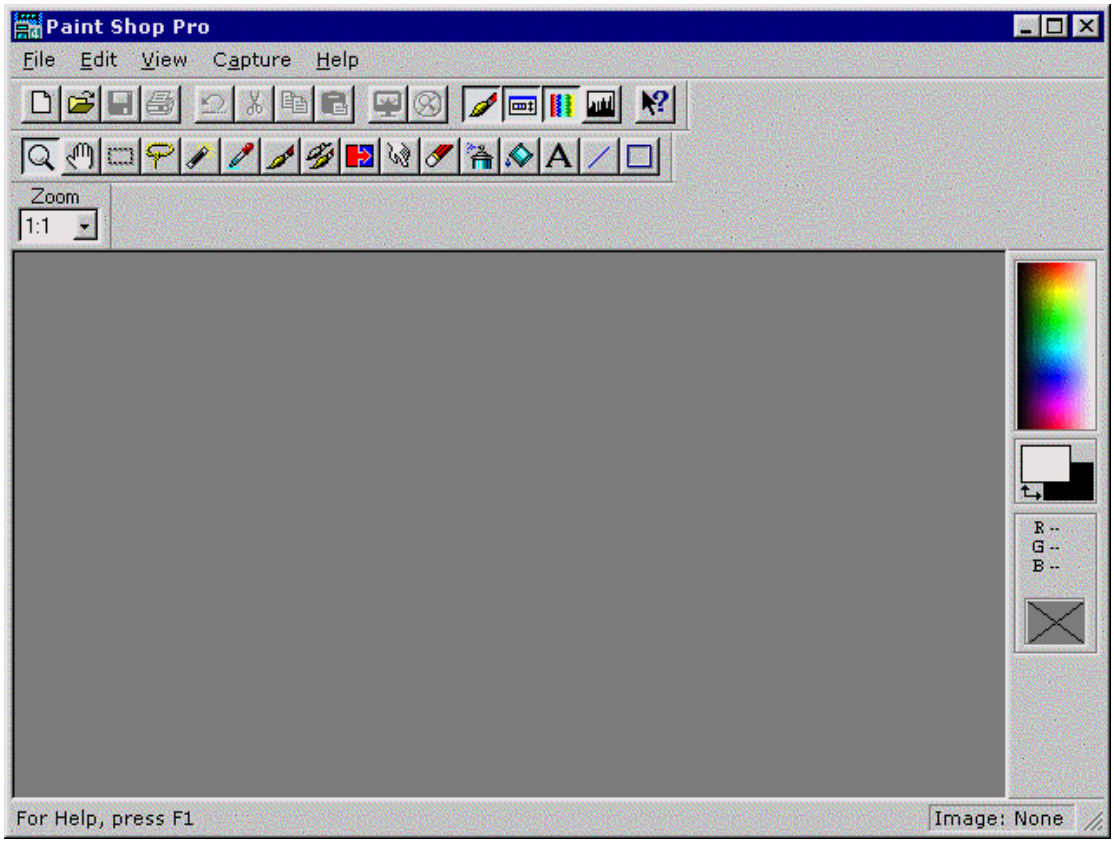
Voit imuroida itsellesi **Paint Shop Pro**n sen kotisivuilta osoitteesta **`http://www.jasc.com`** tai jostain Internetin ohjelmistoarkistosta.

Yksi suosituimpia maksullisia ohjelmia on nimeltään **`Adobe Photoshop`**, jonka voit ostaa lähimmästä tietokoneohjelmia myyvästä liikkeestä. Vanhemmat **Photoshop**in versiot tarvitsevat **gif**-kuvien tallentamiseen apuohjelman, jonka voi imuroida maksutta Adoben kotisivulta osoitteesta **`http://www.adobe.com`**

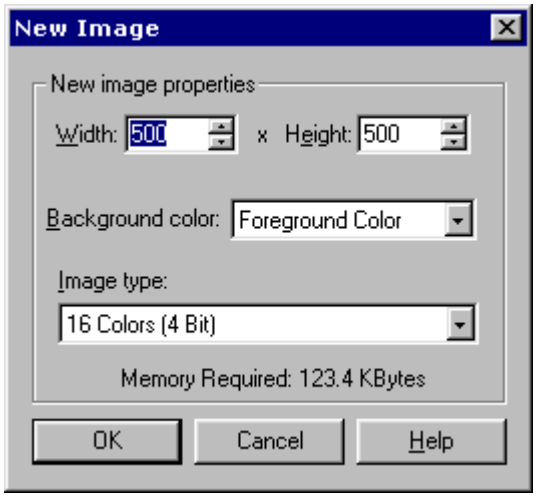
Tässä luvussa näkyvät esimerkit ovat **Paint Shop Pro** -ohjelmasta.

PIIRTÄMINEN

Kun käynnistät **Paint Shop Pro**n, kuvaruudullasi näkyy ohjelman tyhjä aloitusruutu.



Kun alat tekemään grafiikkaa sivuillesi, ensimmäinen toimenpide onkin aloittaa uusi piirros. Tämä tapahtuu valitsemalla **File**-valikosta toiminto **New**. Saat ruudullesi ikkunan, jossa ohjelma kysyy sinulta tietoja kuvan koosta ja värimäärästä:

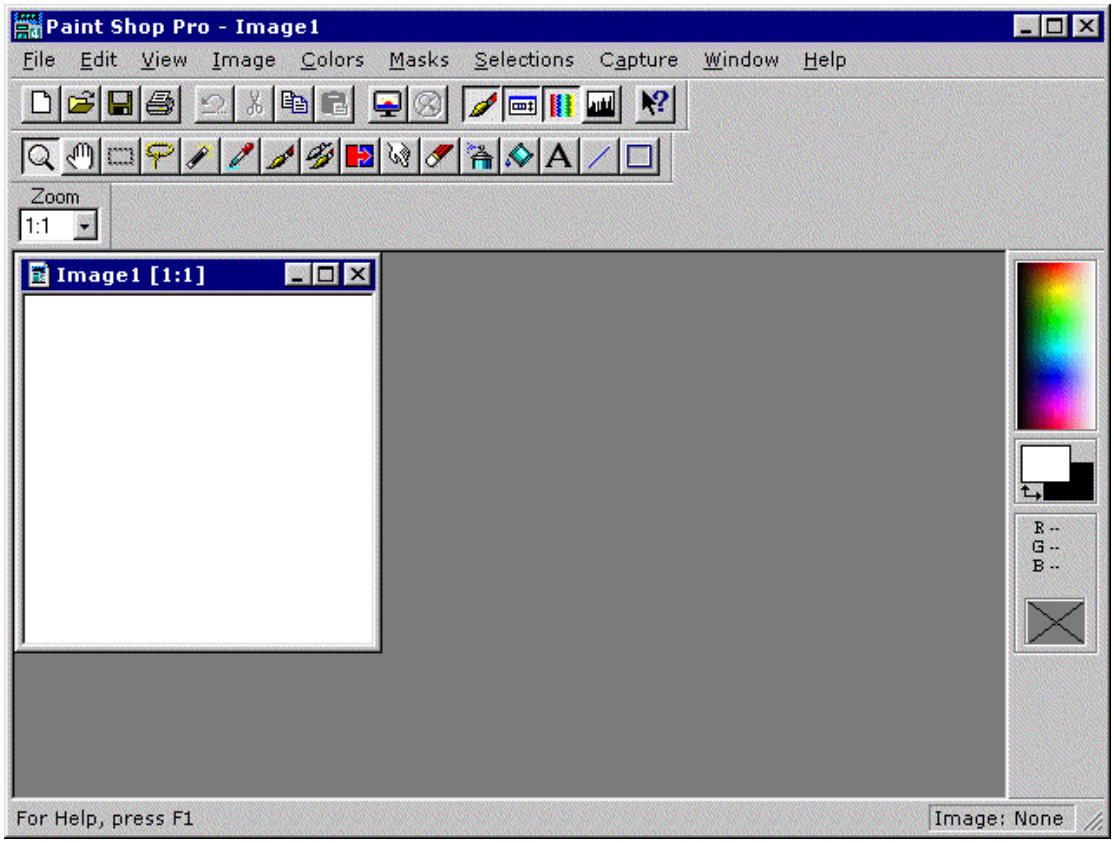


Kuvan koko vaikuttaa suoraan siihen, minkä kokoisena se näkyy selaimesi kuvaruudulla. Aloita ensimmäinen kuvasi antamalla sen leveydeksi (**Width**) 200 ja korkeudeksi (**Height**) niin ikään 200. Tämä tekee kuvasta noin 200 x 200 pikselin kokoisen. Voit myöhemmin halutessasi suurentaa tai pienentää piirrosalustan kokoa.

Voit valita kuvasi taustavärin (piirrosalustan, tai "kankaan" värin) valikosta **Background color**. Ota väriksi vaikkapa valkoinen (**White**). Kuvan värimäärä vaikuttaa siihen, miten monta väriä voit käyttää yhtäaikaa piirtäessäsi kuvaa. Se vaikuttaa myös siihen, miten suurikokoinen kuvatiedostosta tulee ja siihen, miten kauan sen lataaminen kestää.

Valitse kuvasi värimääräksi **256 Colors (8 bit)**. Tämä on tavanomainen värimäärä tehtäessä grafiikkaa WWW-sivuille. Voit myöhemmin halutessasi muokata kuvasi käyttämää värimäärää.

Kun olet antanut halutut tiedot, saat näytöllesi uuden tyhjän alustan, jolle voit aloittaa piirtämisen:



Kuvaruudun vasemmassa reunassa näet käytössäsi olevan väripaletin ja kuvaruudun yläosassa kaksi työkalupalkkia, joista voit valita erilaisia piir-

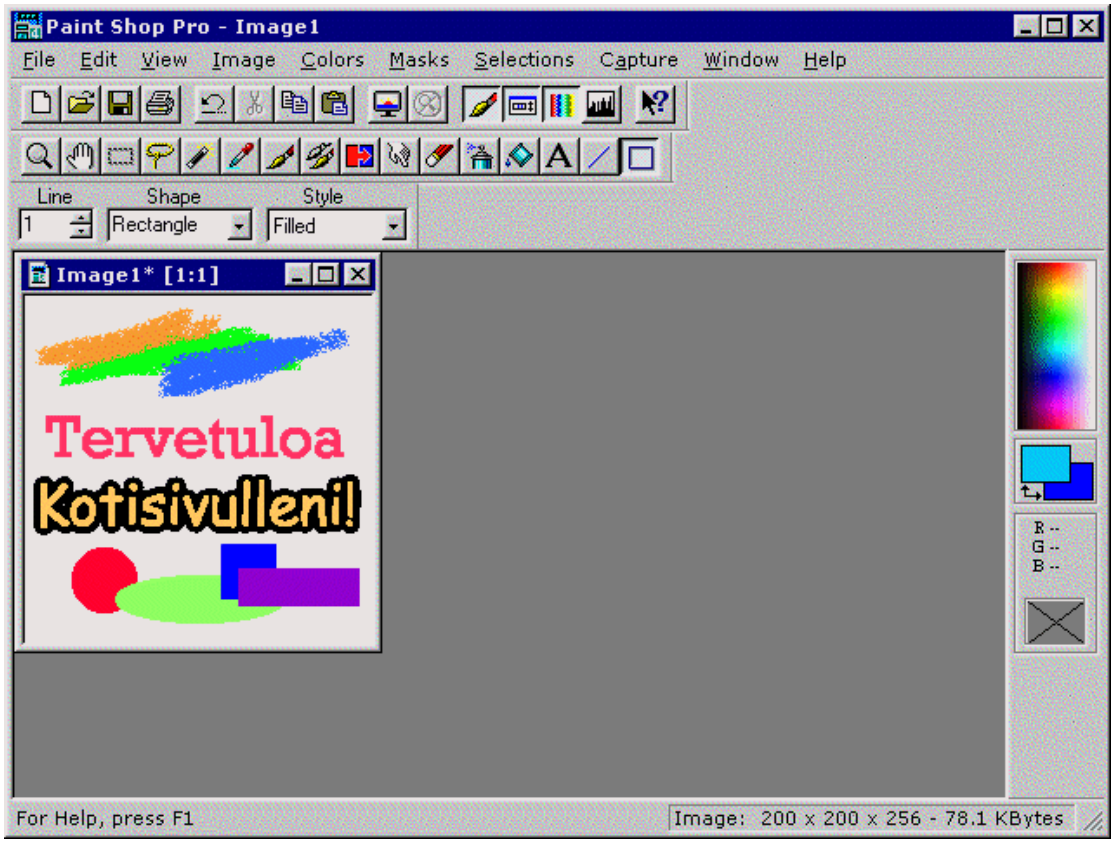
rostyökaluja. Kun viet hiiresi osoittimen työkaluja kuvaavien ikonien päälle, näet kursorin vieressä työkalun nimen.

Kokeile piirtää kotisivullesi sopiva kuva käyttäen eri työkaluja. Voit piirtää vapaita kuvioita pensselityökalulla (**Paintbrush**), nelikulmioita ja ympyröitä muototyökalulla (**Shapes**) ja lisätä tekstiä tekstityökalulla (**Text**).

Väripaletissa on käytössä kaksi väriä, jotka on ohjelmoitu hiiren nappeihin. Väripaletin alapuolella on kaksi laatikkoa, joista näet mitä väriä nappeihin on ohjelmoitu. Päällimmäinen, kokonaan näkyvä laatikko sisältää edustavärin (kuvassa valkoinen). Sitä voit piirtää hiiren vasemmalla napilla. Edustavärin takana osittain piilossa oleva laatikko näyttää taustavärin (kuvassa musta), jota voit piirtää hiiren oikeanpuoleisella napilla.

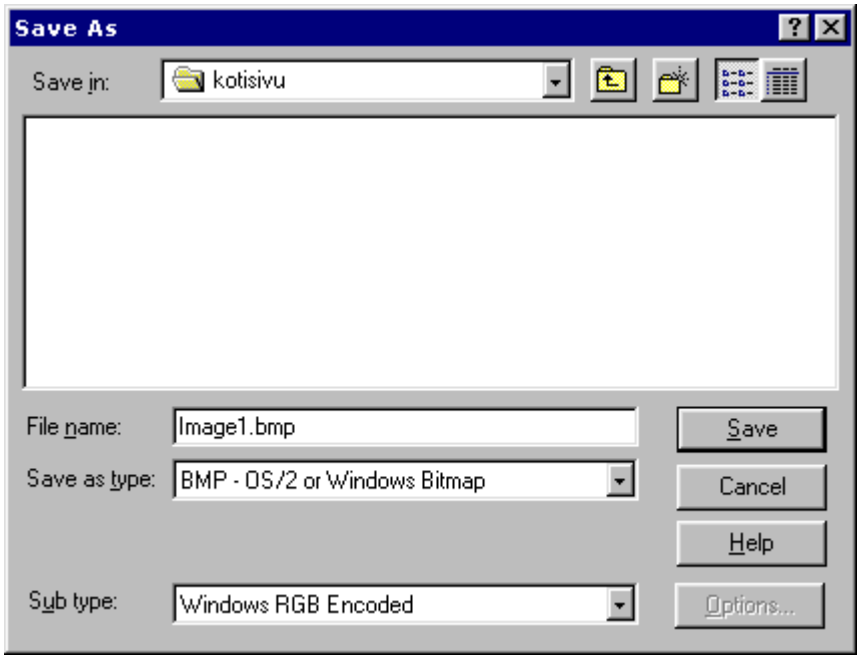
Tutustu myös **Paint Shop Pro**n **Help**-toimintoon, jonka saat **Help**-valikosta tai painamalla F1-näppäintä. Sieltä löydät ohjeet erilaisten toimintojen ja työkalujen käyttämiseen.

Piirrä kuva, jonka haluat näkyviin kotisivullesi:



TALLENTAMINEN

Kun kuvasi on valmis, on aika tallentaa se. Kuva tallennetaan valitsemalla **File**-valikosta toiminto **Save As**. Saat ruudullesi näkyviin tallennusikkunan:



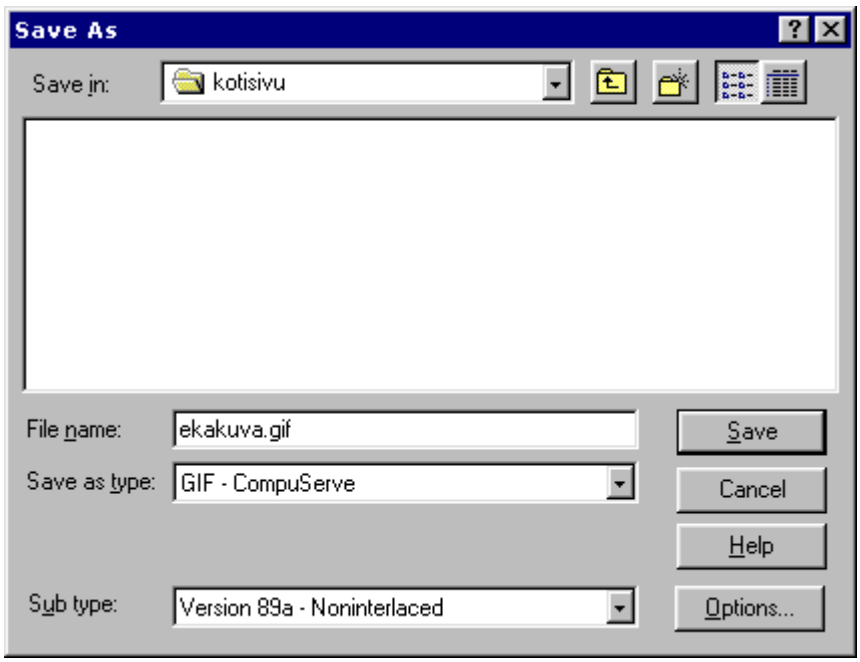
(Ikkuna saattaa olla toisen näköinen, jos et käytä Windows 95 -käyttöjärjestelmää)

Tässä ikkunassa voit kertoa minne kuvasi tallennetaan, sen nimen ja sen, missä muodossa haluat tallentaa kuvasi. Etsi ensimmäiseksi se hakemisto, jonne olet tallentanut aiemmin tekemäsi HTML-tiedostot. Kuva tallennetaan niiden kanssa samaan hakemistoon. Huomaa, että **Paint Shop Pro** ei näytä hakemistossa olevia HTML-tiedostoja, vaan keskellä näkyvällä alueella näkyvät vain muut hakemistossa olevat kuvat.

Seuraavaksi voit antaa kuvallesi nimen. Kirjoita tiedoston nimi pienillä kirjaimilla äläkä käytä siinä välilyöntejä tai ääkkösiä (vaikka tietokoneessasi niitä voisi muuten käyttääkin). Anna nimelle tarkennin **`.gif`**, eli esimerkiksi **`ekakuva.gif`**.

Seuraavaksi valitaan kuvan tallennusmuoto. WWW-sivuille voi sijoittaa joko **gif**- tai **jpeg**-muodossa olevia kuvia. Tyyppi valitaan kohdasta **Save as type**. Valitse tästä valikosta kuvasi tallennusmuodoksi "**GIF - CompuServe**".

Kun olet antanut nämä tiedot, voit tallentaa kuvasi painamalla **OK**-painiketta:



KUVAT WWW-SIVULLA

Kun kuvasi on piirretty ja tallennettu, on aika siirtyä takaisin HTML-koodin pariin. Jos olet sammuttanut välillä HTML-editorisi, käynnistä se uudelleen ja lataa siihen aikaisemmin muokkaamasi tiedosto, **`kotisivu.htm`** (jos olet tallentanut sivusi jollain muulla nimellä, käytä sitä nimeä).

Kuva liitetään HTML-tiedoston sisään komennolla **`<img>`**. Tämä komento yksinään ei tee vielä mitään, vaan se vaatii lisäattribuutin **`src`**, joka kertoo mistä kuva löytyy. Lopullinen linkityskomento näyttää tältä:

```
<img src="kuvan tiedostonnimi">
```

Lainausmerkkien väliin sijoitetaan se nimi, jolla kuva on tallennettu. Kuvakomento laitetaan HTML-tiedostossa siihen kohtaan, johon haluat kuvan ilmestyvän.

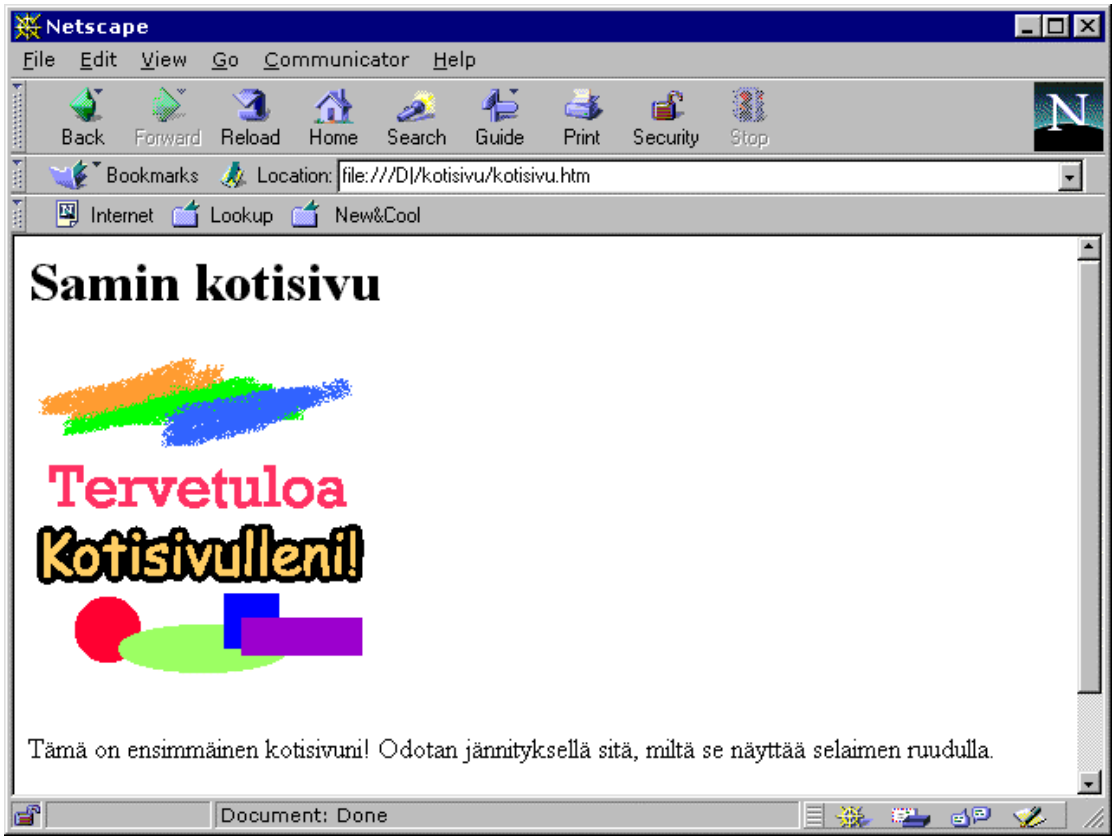
Kun joku lataa myöhemmin sivuasi, hänen selaimensa tulostaa HTML-tiedostossa olevaa tekstiä kuvaruudulle komentojen mukaan. Kun selain saapuu kohtaan, jossa kuvakomento on, se etsii komennossa kerrotun kuvatiedoston ja tulostaa sen selaimen ruudulle siihen kohtaan, missä komento oli. Kuvia sisältävät sivut kootaan siis paloista: teksti ja kuvat sijait-

sevat tietokoneen kovalevyllä eri tiedostoissa, mutta kun sivua luetaan selaimella, ne kerätään yhdeksi kokonaiseksi sivuksi lukijan näytölle.

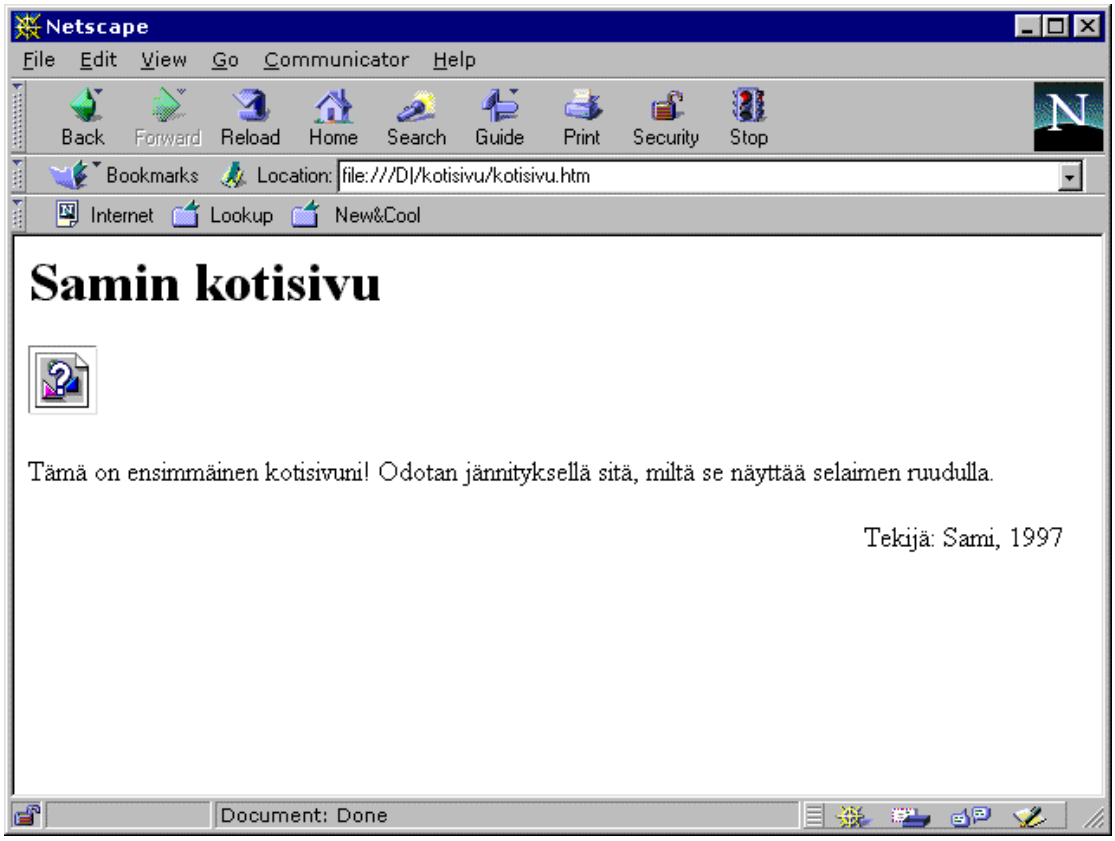
Kokeillaan tätä käytännössä: lisää aiemmin tekemällesi kotisivulle komento, joka liittää siihen äsken tekemäsi kuvan:

```
<h1>Samin kotisivu</h1>
<img src="ekakuva.gif">
<p>Tämä on ensimmäinen kotisivuni! Odotan
jännityksellä sitä, miltä se näyttää selaimen
ruudulla.</p>
<p align=right>Tekijä: Sami, 1997</p>
```

Tallenna tiedosto, ja kokeile katsella sitä selaimessasi. Tuliko kuva näkyviin otsikon alle?



Joskus saattaa käydä niin, että kuva ei tulekaan näkyviin, vaan sen tilalla näkyy epäonnistuneen kuvan symboli:



Jos kuvan lisäys epäonnistuu, siihen voi olla monta syytä. Alla on lista tavallisimmista syistä ja korjauskeinoista:

- Väärä tiedoston nimi. Tarkista että kirjoittamasi HTML-komennon lainausmerkkien välissä on sama nimi kuin nimi, jolla tallensit tiedoston. Tarkista myös, että tiedoston nimen tarkennin on **`.gif`**. Vältä käyttämästä tiedoston nimessä ääkkösiä (Å-, Ä- tai Ö-kirjaimia ja muitakin erikoismerkkejä), sillä ohjelmat ja tietokoneet ovat yleensä englanninkielisissä maissa tehtyjä eivätkä välttämättä ymmärrä muunmaalaisia merkistöjä. Vältä myös välilyöntejä, sillä etenkään vanhemmat tietokoneet eivät sulata tiedostojen nimissä olevia välilyöntejä. Kirjoita tiedostojen nimet aina pienillä kirjaimilla - Windows-koneet eivät merkkien koosta välitä, mutta palvelinkoneet saattavat pitää esimerkiksi nimiä "Kuva" ja "kuva" eri niminä!

- Väärä hakemisto. Tarkista, että tallensit kuvan samaan hakemistoon, jossa HTML-tiedostosi (esimerkissä **`kotisivu.htm`**) sijaitsee.

- Kirjoitusvirhe HTML-komennossa. Tarkista ettei komennosta puutu toista lainausmerkkiä tai komennon aloitus- tai lopetusmerkkiä. Tarkista myös kirjoititko itse komennon ja attribuutin oikein: **`img src`**.

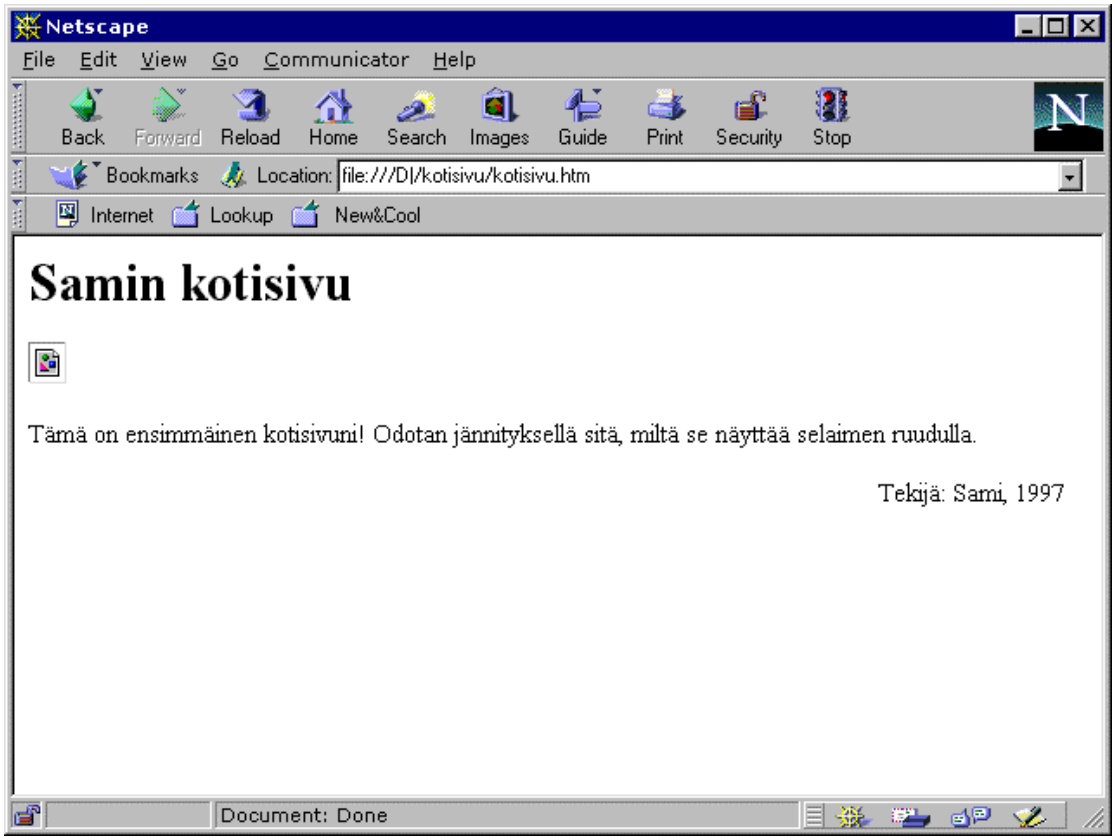
- Väärässä muodossa tallennettu kuva. Muista tallentaa kuva **gif**- tai **jpeg**-muodossa. Selaimet eivät ymmärrä (vielä) muita kuvaformaatteja.

- Tiedoston oikeudet ovat väärin. Tämä saattaa tapahtua silloin, jos olet jo siirtänyt tiedostosi operaattorin palvelinkoneelle. Lue lisätietoja tiedoston siirtoa käsittelevästä kappaleesta.

KUVIEN SELITYSTEKSTIT

Kaikki WWW-sivujen lukijat eivät pysty katsomaan kuvia omalla koneellaan. Tällaisia käyttäjiä varten sivuille kannattaa lisätä kuvien selitystekstit.

Jokaisessa selaimessa on mahdollista kytkeä kuvien lataus pois päältä. Tällöin selain jättää kaikki kuvat lataamatta ja näyttää niiden tilalla pelkän kuvasymbolin:



Kun kuvia ei ladata, WWW-sivuilla surffailu käy paljon nopeammin. Monet modeemin yli Internet-yhteyksiä käyttävät kytkevät usein kuvat pois päältä nopeuttaakseen selaimensa toimintaa.

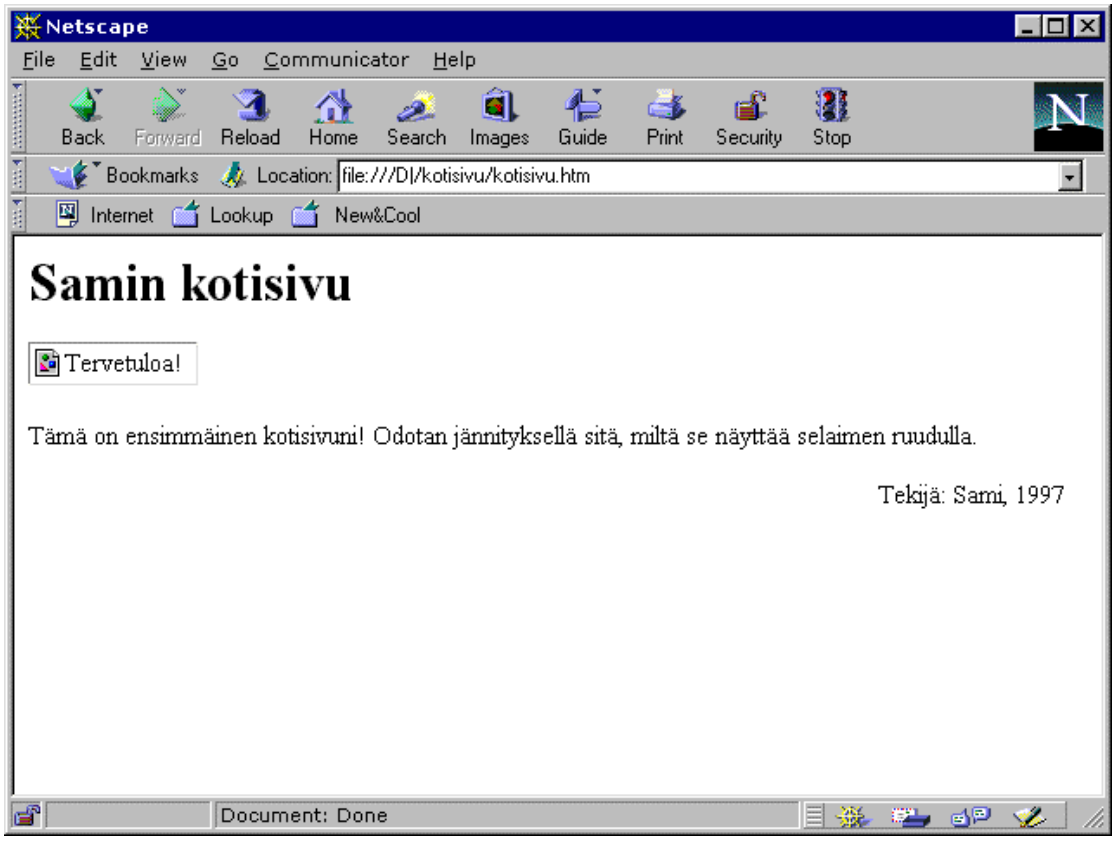
Tällöin kuitenkin lukijat menettävät paljon sivun sisällöstä. Jos sivulla olevien kuvien toiminnalla on tärkeää merkitystä, käyttäjät eivät voi nähdä mitä kuvien on tarkoitus esittää.

Tämän vuoksi WWW-sivuille sijoitettaviin kuviin on mahdollista lisätä selitysteksti. Selitysteksti lisätään kuvakomennon sisään attribuutilla **alt="*teksti*"**. Lainausmerkkien väliin voit kirjoittaa tekstiä, joka tulee näkyviin lukijalle silloin kun hän ei syystä tai toisesta pysty katselemaan sivullasi olevia kuvia. Tekstissä ei voi käyttää HTML-komentoja.

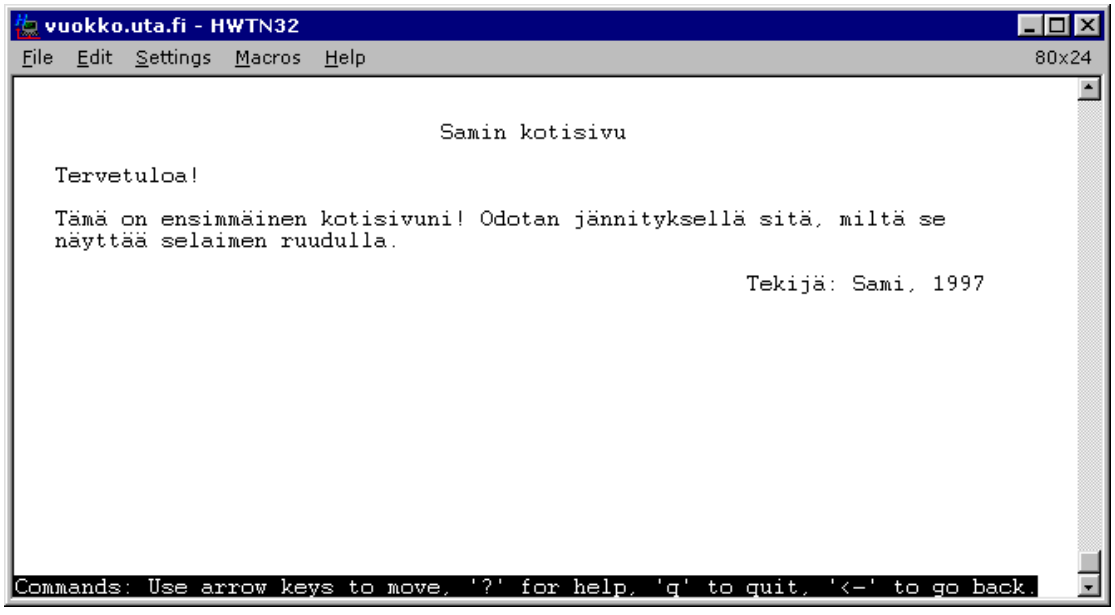
Kokeile muuttaa kotisivuasi siten, että sivullasi oleva kuva sisältää myös selitystekstin:

```
<h1>Samin kotisivu</h1>
<img src="ekakuva.gif" alt="Tervetuloa!">
<p>Tämä on ensimmäinen kotisivuni! Odotan
jännityksellä sitä, miltä se näyttää selaimen
ruudulla.</p>
<p align=right>Tekijä: Sami, 1997</p>
```

Nyt kuvien latauksen pois kytkenyt lukija näkee sivullesi saapuessaan lataamatonta kuvaa esittävän ikonin vieressä selitystekstin, joka kertoo hänelle mitä kuva esittää:



Lukija saattaa käyttää myös sellaista tietokonetta, päätettä tai ohjelmaa, joka ei osaa näyttää lainkaan kuvia (**telnet**-yhteys, tekstipääte) tai ainoastaan rajatusti kuvia (**Nokia Communicator**). Myös näiden laitteiden käyttäjille selitysteksteistä on paljon hyötyä. Esimerkiksi tekstipäätteellä sivujasi katsova näkee kuvan sijasta sen selitystekstin:



Jos kuvassa ei olisi ollut selitystekstiä, tekstipäätettä käyttävä lukija olisi nähnyt sen tilalla tekstin **[IMAGE]** tai **[INLINE]**.

KUVIEN KANSSA TYÖSKENTELY

Kun olet saanut kuvan toimimaan ensimmäistä kertaa, haluat varmaankin muokata sitä tai tehdä uusia kuvia. Kuvien kanssa työskentely on samanlaista kuin HTML-tiedostojenkin kanssa: kuvia piirretään ja käsitellään kuvankäsittelyohjelmassa ja sitten ne tallennetaan. Tämän jälkeen siirrytään selaimeen ja napsautetaan **Reload**-painiketta, jolloin selain lukee muuttuneen kuvan kovalevyltä ja tuo sen näkyviin ruudulle.

Jos haluat lisätä sivulle uusia kuvia tai muuttaa vanhojen kuvien paikkoja, siirryt HTML-editoriisi, jossa siirrät kuvien HTML-komennot oikeisiin paikkoihin tai lisäät joukkoon uusia kuvia tuovia komentoja. Yksi komento tuo ruudulle aina yhden kuvan, joten jokaista kuvaa varten tarvitaan oma komento. Jokainen kuva sijaitsee niin ikään omassa tiedostossaan. HTML-tiedoston tallentamisen jälkeen voit käydä taas katselemassa muutoksia selaimessasi (älä unohda painaa **Reload**-painiketta!).

Kun olet tehnyt sivuja jotka sisältävät monia kuvia, huomaat nopeasti että hakemistosi täyttyy monista tiedostoista. Selaimen ruudulla yhtenä sivuna näkyvä sivu saattaa koostua kymmenistä eri tiedostoista: HTML-tiedostosta ja useasta sivulle tuotavasta kuvasta. Voi olla hyvä idea pitää kirjaa kuhunkin sivuun kuuluvista kuvatiedostoista, jotta et mene sekaisin tiedostojen kanssa.

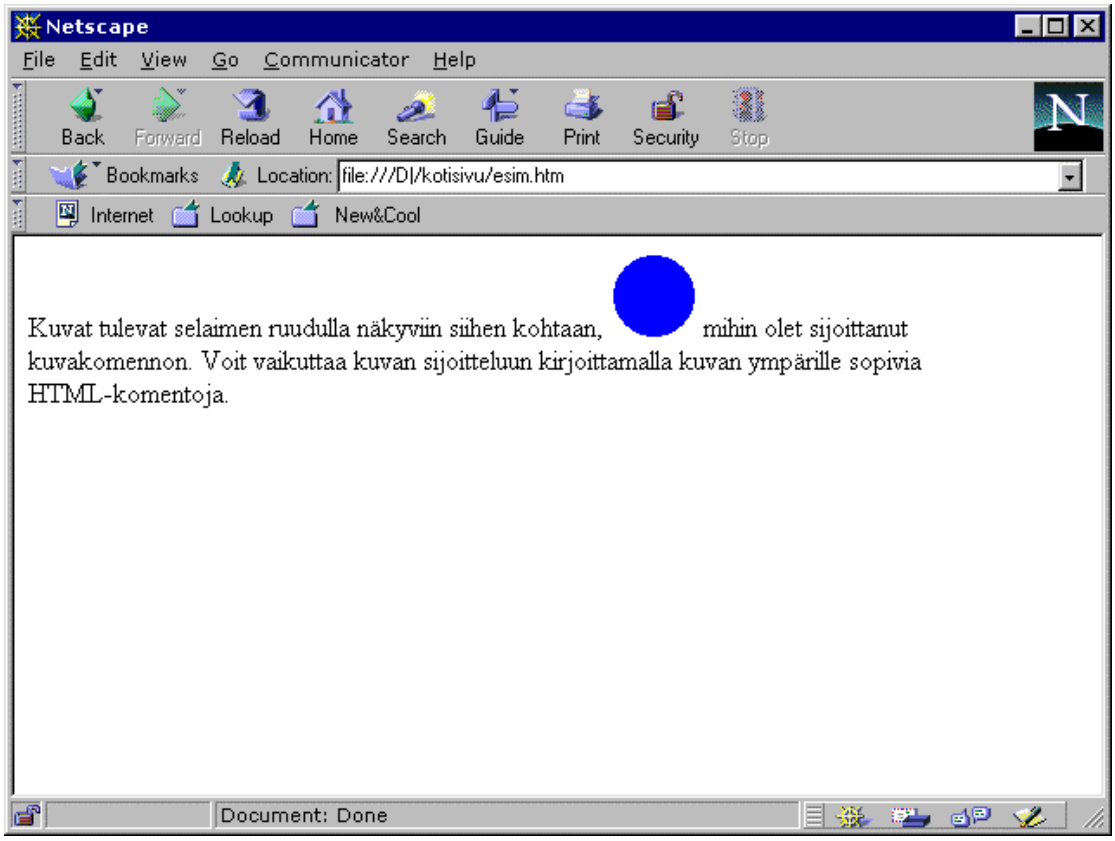
KUVIEN SIJOITTELU

Kuvat tulevat selaimen ruudulla näkyviin siihen kohtaan, mihin olet sijoittanut kuvakomennon. Voit vaikuttaa kuvan sijoitteluun kirjoittamalla kuvan ympärille sopivia HTML-komentoja.

Jos kirjoitat kuvakomennon keskelle tekstiriviä, myös itse kuva tulee näkyviin tekstin keskelle:

```
<p> Kuvat tulevat selaimen ruudulla näkyviin siihen
kohtaan, <img src="pallo.gif"> mihin olet sijoittanut
kuvakomennon. Voit vaikuttaa kuvan sijoitteluun
kirjoittamalla kuvan ympärille sopivia HTML-komentoja.
```

Sama selaimessa:



Muut tekstirivit antavat automaattisesti tilaa kuvalle, joten mitä korkeampi kuva on, sitä suurempi rivinväli kuvan kohdalle muodostuu.

Jos haluat sijoittaa kuvan omalle rivilleen erilleen tekstistä, voit laittaa kuvakomennon eteen kappaleenvaihtokomennon. Kuvan jälkeen tuleva teksti siirtyy niin ikään omalle rivilleen kun laitat tekstin alkuun kappaleenvaihtokomennon.

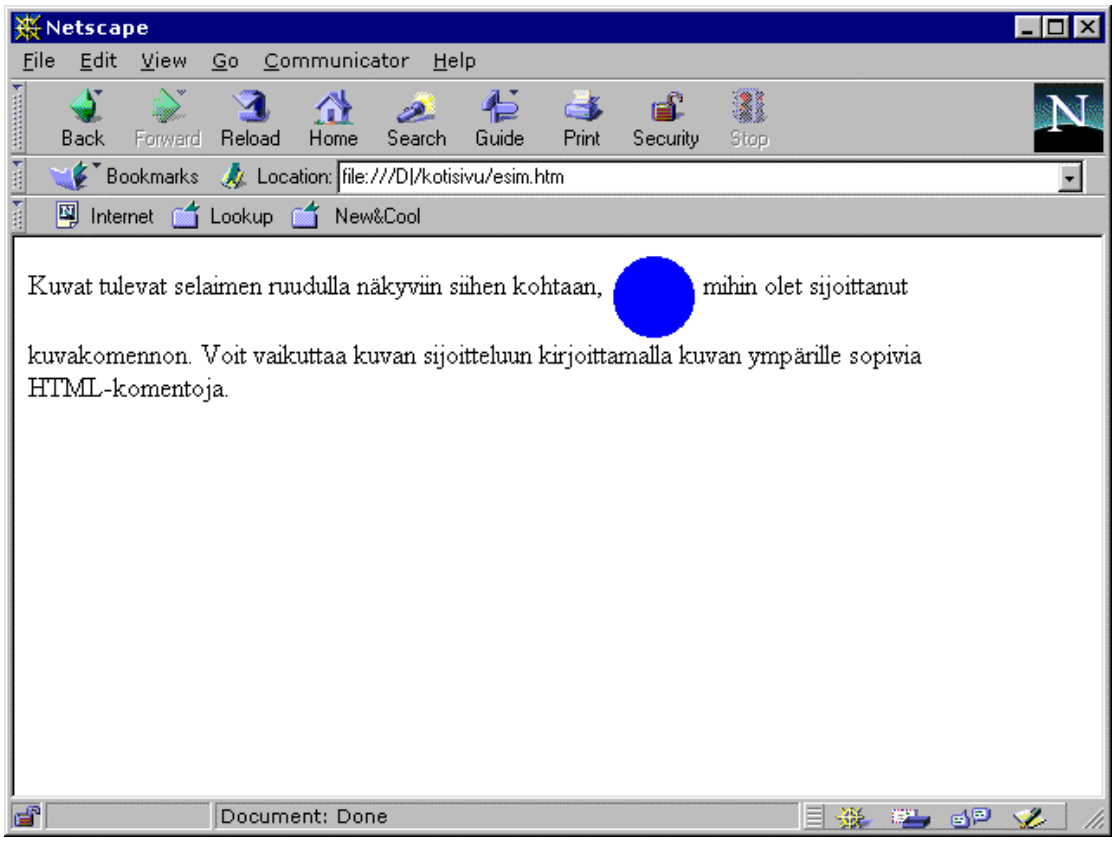
Jos haluat sijoittaa kuvan lähemmäs tekstiä, voit käyttää kappaleenvaihtojen sijasta rivinvaihtokomentoja.

Kuviin on mahdollista lisätä attribuutteja, jotka kertovat sen, miten kuva suhtautuu sen ympärillä olevaan tekstiin. Normaalisti rivin keskellä oleva kuva "kasvaa riviltä ylöspäin", kuten yllä näkyvässä esimerkissä. Kuvan alareuna on siis samalla tasalla kuin tekstin alareuna.

Voit vaihtaa tätä sijoittelua attribuuteilla **`align=middle`** ja **`align=top`**. Muutetaan hiukan aikaisempaa esimerkkiä:

```
<p> Kuvat tulevat selaimen ruudulla näkyviin siihen
kohtaan, <img src="pallo.gif" align=middle> mihin olet
sijoittanut kuvakomennon. Voit vaikuttaa kuvan
sijoitteluun kirjoittamalla kuvan ympärille sopivia
HTML-komentoja.
```

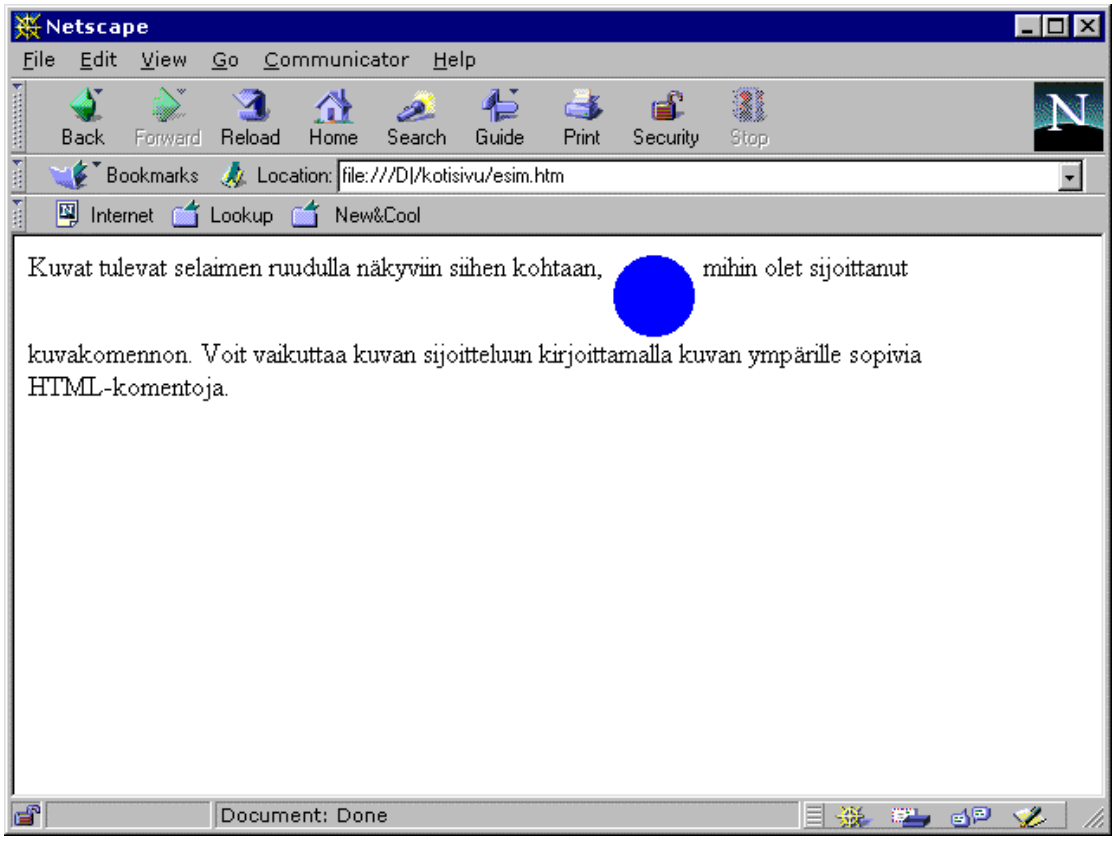
Huomaa, kuinka tämä muutos vaikuttaa selaimen ruudulla:



Tekstin alareuna onkin nyt samalla tasalla kuvan keskikohdan kanssa. Jos haluat sijoittaa kuvan yläreunan samalle tasalle kuin tekstin yläreunan, eli saada kuvan "roikkumaan tekstiriviltä alaspäin", voit käyttää siihen attribuuttia **`align=top`**:

```
<p> Kuvat tulevat selaimen ruudulla näkyviin siihen
kohtaan, <img src="pallo.gif" align=top> mihin olet
sijoittanut kuvakomennon. Voit vaikuttaa kuvan
sijoitteluun kirjoittamalla kuvan ympärille sopivia
HTML-komentoja.
```

Sama selaimessa:



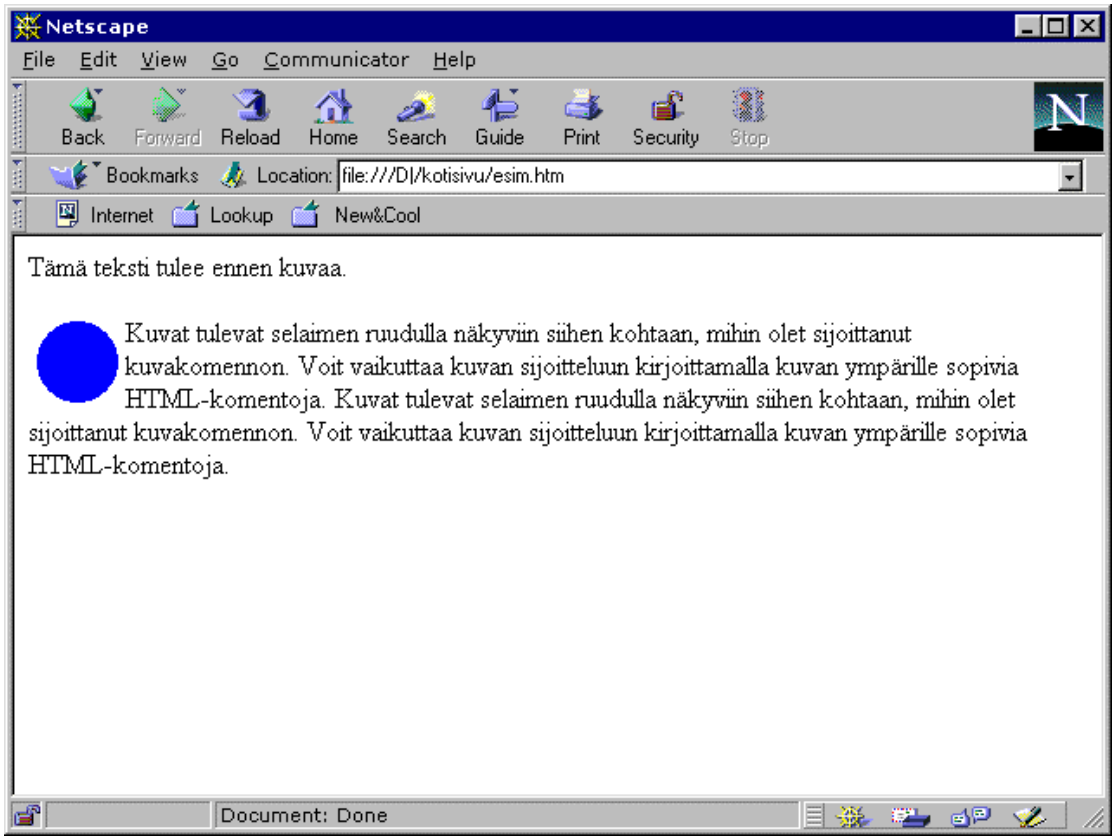
Normaalisti kuvan viereen sopii vain yksi rivi tekstiä. Edelliset ja seuraavat rivit tulevat näkyviin kuvan ylä- ja alapuolille. Attribuuteille **`align=left`** ja **`align=right`** voit kuitenkin muuttaa kuvan toimintaa siten, että kuvan viereen tulee useampia tekstirivejä.

Näitä attribuutteja käytettäessä kuva siirtyy aina selaimen reunaan. Kaikki komennon jälkeen tuleva teksti siirtyy kulkemaan kuvan toiselta puolelta. Tekstiä kulkee kuvan vieressä niin paljon kuin kuvaa riittää, jonka jälkeen teksti siirtyy taas kulkemaan normaalisti.

Esimerkki tekstin kierrättämisestä kuvan ympäri:

```
<p>Tämä teksti tulee ennen kuvaa.
<p><img src="pallo.gif" align=left> Kuvat tulevat
selaimen ruudulla näkyviin siihen kohtaan, mihin olet
sijoittanut kuvakomennon. Voit vaikuttaa kuvan
sijoitteluun kirjoittamalla kuvan ympärille sopivia
HTML-komentoja. Kuvat tulevat selaimen ruudulla
näkyviin siihen kohtaan, mihin olet sijoittanut
kuvakomennon. Voit vaikuttaa kuvan sijoitteluun
kirjoittamalla kuvan ympärille sopivia HTML-komentoja.
```

Katsoessasi esimerkkiä selaimessa huomaat, kuinka pallon kuva jää vasempaan reunaan ja komennon jälkeen tuleva teksti kiertää kuvan:



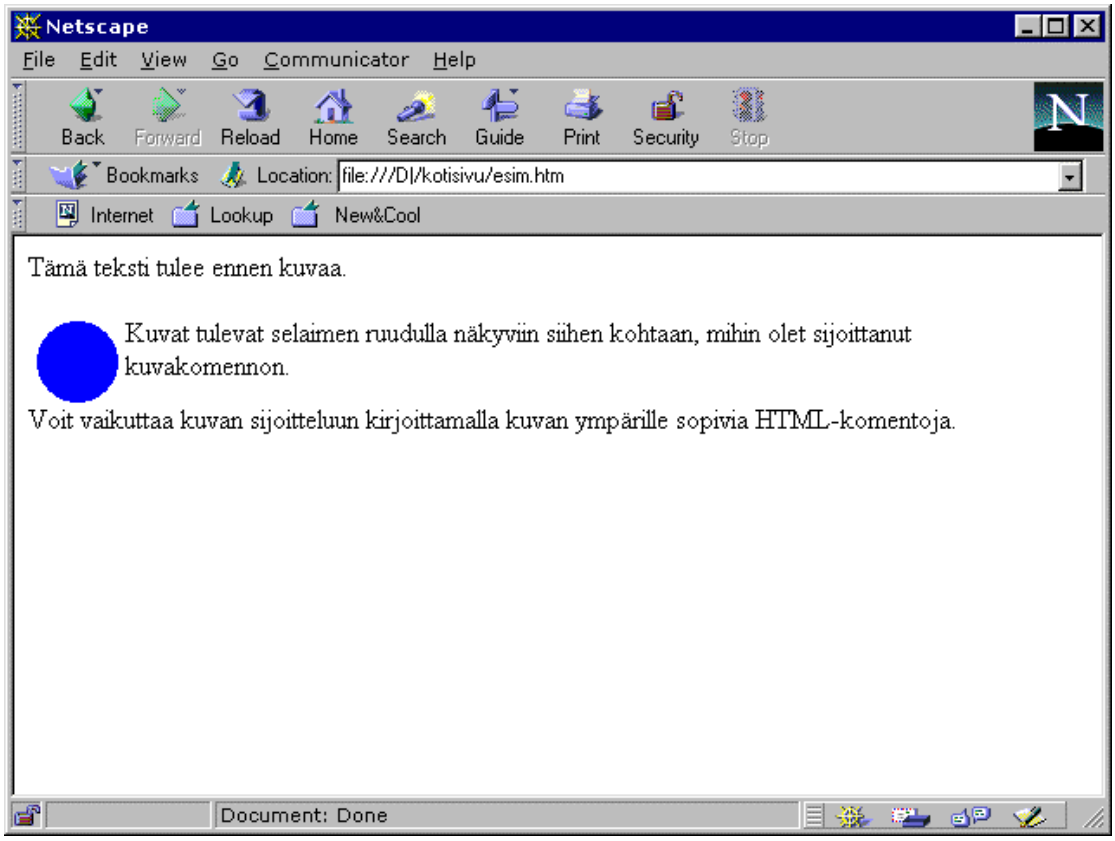
Attribuutti **`align=right`** saa kuvan käyttäytymään samalla tavalla, mutta kuva siirtyykin selaimen ruudun oikeanpuoleiseen reunaan ja teksti kiertää sen vasemmalta.

Jos haluat rajoittaa kuvan vieressä näkyvän tekstin määrää, voit sijoittaa tekstiin lisäattribuutilla varustetun rivinvaihtokomennon, joka saa tekstin kierrätyksen loppumaan.

Jos kuva on sijoitettu vasempaan reunaan (kuten yllänäkyvässä esimerkissä), voit lopettaa tekstin kierrätyksen HTML-komennolla **`<br clear=left>`**. Kaikki tämän komennon jälkeen tuleva teksti siirtyy kuvan alapuolelle, siihen kohtaan, jossa vasen reuna on jälleen "normaalissa kohdassa":

```
<p>Tämä teksti tulee ennen kuvaa.
<p><img src="pallo.gif" align=left> Kuvat tulevat
selaimen ruudulla näkyviin siihen kohtaan, mihin olet
sijoittanut kuvakomennon. <br clear=left> Voit
vaikuttaa kuvan sijoitteluun kirjoittamalla kuvan
ympärille sopivia HTML-komentoja.
```

Selaimessa näet, miten tämä erikoisempi rivinvaihtokomento vaikuttaa:



Jos kuvassa on käytetty attribuuttia **`align=right`**, sitä vastaava rivinvaihtokomento on **`<br clear=right>`**. Voit käyttää myös yleisluontoisempaa rivinvaihtokomentoa **`<br clear=both>`**, jolloin teksti siirtyy alaspäin kunnes kummassakaan reunassa ei ole kuvaa.

KUVAN LINKITTÄMINEN MUUALTA VERKOSTA

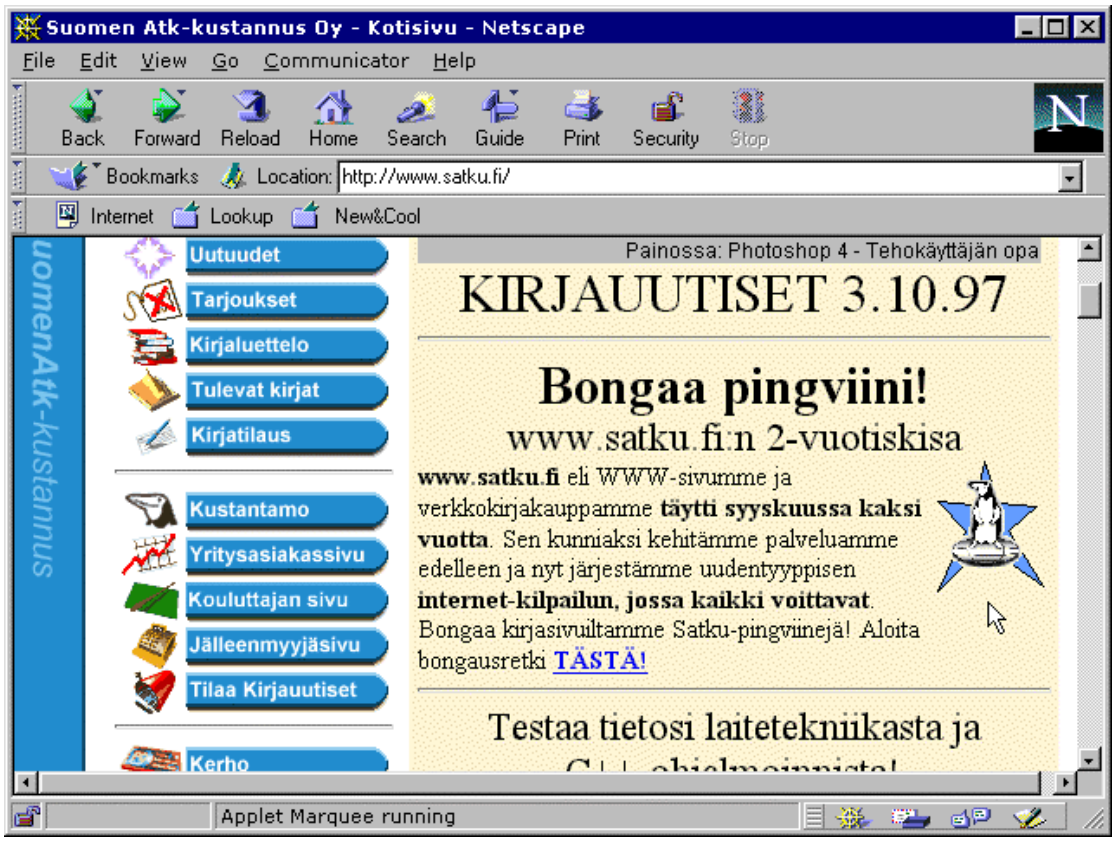
WWW-sivulla sijaitsevan kuvan ei ole pakko sijaita samassa hakemistossa itse HTML-tiedoston kanssa. Itse asiassa kuvan ei tarvitse sijaita edes samassa tietokoneessa!

Olet aiemmin huomannut, että jokaisella Internetissä olevalla WWW-sivulla on oma osoite, joka alkaa yleensä tekstillä **`http://`**. Tämä pätee varsinaisten sivujen lisäksi myös kuviin. Jokaisella verkossa olevalla kuvalla on oma URL-osoite aivan kuten WWW-sivuillakin. Kuvakomennon sisään on mahdollista sijoittaa tiedoston nimen sijasta myös kuvan verkko-osoite.

Kun selain alkaa näyttää WWW-sivua lukijan kuvaruudulla, se etsii kuvakomennoissa mainitut tiedostot ja tuo niiden sisältämän kuvan oikeaan

paikkaan sivun keskelle. Jos selain löytääkin kuvakomennosta tiedoston nimen sijasta verkko-osoitteen, se käy hakemassa kyseisen kuvan verkosta ja näyttää sen sitten kuvaruudullasi.

Oletetaan vaikka, että haluat käyttää omalla sivullasi Suomen ATK-kustannuksen sivuilta löytyvää pingviini-logoa:

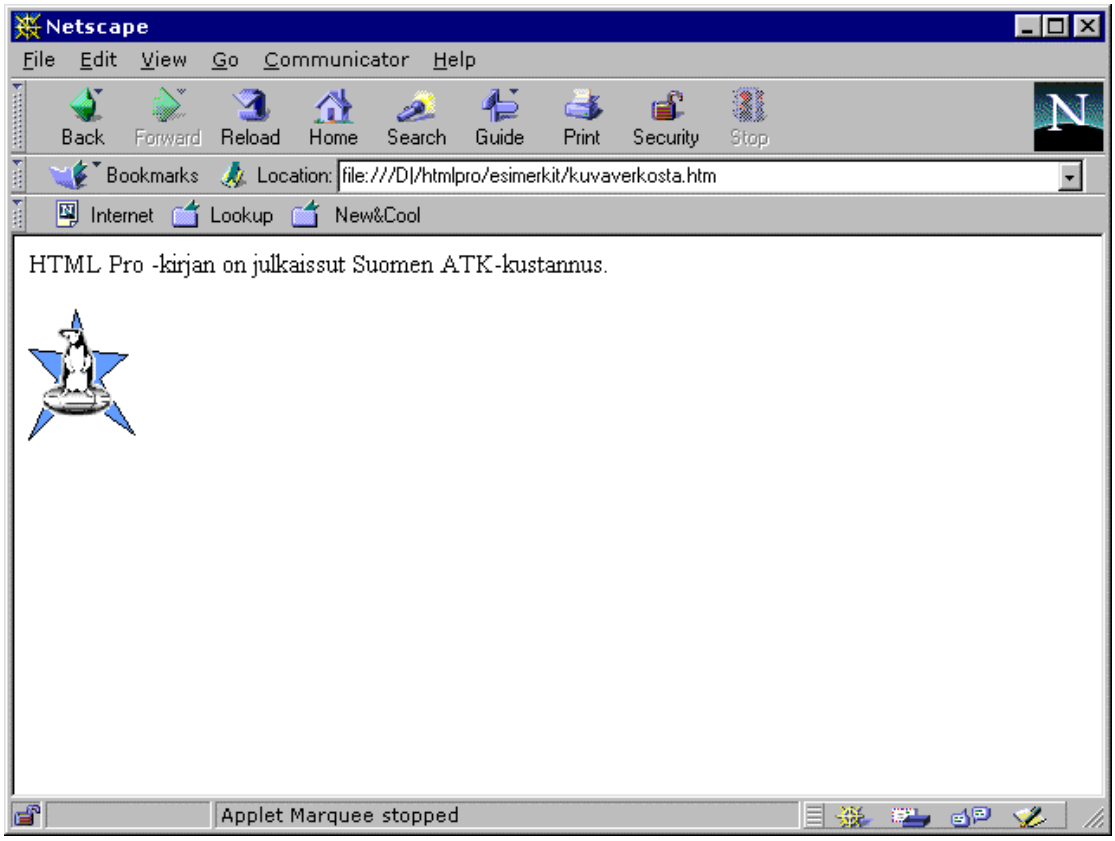


Ensimmäinen tehtäväsi on ottaa selville kuvan osoite. Netscapea käyttäessäsi saat selvitettyä kuvan osoitteen viemällä hiiren osoittimen kuvan päälle ja napsauttamalla hiiren oikeanpuoleista nappia. Ruudullesi avautuu valikko, jossa yhtenä kohtana on **Copy Image Location**:

Kun valitset tuon valikon kohdan, osoittamasi kuvan osoite kopioituu leikepöydälle, josta voit sitten liimata sen HTML-editoriisi kuvakomennon lainausmerkkien väliin:

```
<p>HTML Pro -kirjan on julkaissut Suomen ATK-
kustannus.
<p><img src="http://www.satku.fi/etu_bong.gif">
```

Kun katselet sivua selaimellasi, se käy hakemassa **www.satku.fi** -palvelimesta halutun kuvan ja tuo sen näkyviin sivun sisään haluttuun kohtaan:



Muista kuitenkin, että toisessa palvelimessa sijaitsevien kuvien osoite saattaa muuttua. Silloin myös sinun sivullasi näkyvä kuva lakkaa toimimasta.

Kuvan sisältö saattaa myös vaihtua toiseksi arvaamatta. Jos et osannut odottaa tätä, sivusi saattavat muuttua yllättäen toisen näköisiksi. Voit kuitenkin käyttää tätä myös hyödyksesi. Verkossa on monia palveluita, joissa kuvien sisältö vaihtuu jatkuvasti. Tällaisia sivuja ovat esimerkiksi Etelä-Suomen säätutkakuva sivulla

`http://cumulus.helsinki.fi/tutkakuva.html`

tai useat satelliittikuvat. Myös monet verkkosarjakuvat pitävät päivittäin vaihtuvaa sarjakuvaa samassa osoitteessa.

Kun käytät muiden sivujen kuvia omilla sivuillasi, kysy aina ensin lupaa kuvan käyttämiselle. Kuvan tekijänoikeuksien omistaja tai sivujen tekijä ei välttämättä halua kuvaansa käytettävän muilla verkkosivuilla, etenkään jos oikeuksien haltija kerää kyseisellä kuvalla kävijöitä omille sivuilleen.

KUVIEN KOPIOIMINEN VERKOSTA

Netscape tarjoaa mahdollisuuden kopioida verkosta löytyvä kuva omaan tietokoneeseen. Jos et halua käyttää verkosta löytyvää kuvaa sellaisenaan, vaan haluat ottaa kuvasta itsellesi kopion, voit käyttää Netscapen **Save Image** -toimintoa.

Saat toiminnon esiin napsauttamalla hiiren oikeanpuoleista nappia. Ruudullesi tulee näkyviin edellisestä esimerkistä tuttu valikko. Valitse tällä kertaa valikosta toiminto **Save Image As**. Saat seuraavaksi ruudullesi tallennusikkunan, josta voit tallentaa kuvan haluamaasi hakemistoon:

Tallennusikkunassa on valmiiksi valittuna oikea tallennusmuoto (**gif** tai **jpeg**). Siinä on myös valmiina kuvan nimi siinä muodossa kuin alkuperäisen sivun tekijä on sen tallentanut. Voit vaihtaa nimen tilalle haluamasi nimen, jos haluat.

Kun olet tallentanut kuvan itsellesi, voit käyttää sitä sivuillasi aivan kuten itse piirtämiäsi kuvia.

Muista kuitenkin, että tarvitset kuvan kopioimiseen ja käyttämiseen aina kuvan tekijänoikeuksien omistajan luvan.

KUVIEN TEKIJÄNOIKEUDET

Vaikka verkossa sijaitsevien kuvien kopioiminen ja käyttäminen on WWW:n ansiosta erittäin helppoa, muista kuitenkin että suurin osa verkossa sijait-

sevista kuvista on tekijänoikeuksien alaisia. Tämä tarkoittaa sitä, että tarvitset kuvan käyttämiseen kuvan oikeuksien omistajan luvan.

Verkosta löytyy kuva-arkistoja, joiden kuvat ovat määritelty julkiseksi omaisuudeksi. Näitä kuvia voit käyttää omilla sivuillasi, joko tallentamalla kuvat itsellesi tai sijoittamalla kuvan osoitteen kuvakomentoosi.

LINKIT - TIEDOSTOJEN LIITTÄMINEN TOISIINSA

Linkit ovat koko hypertekstijärjestelmän idea — linkkejä sisältävää tekstiä kutsutaankin hypertekstiksi. Linkkien avulla sivuja yhdistellään toisiinsa suuremmiksi kokonaisuuksiksi.

HTML:n linkkauskomento on **<a>**. Tällä komennolla määritellään linkki muualla sijaitsevaan sivuun. Pelkkä komento **<a>** ei tee vielä itsessään mitään vaan siihen lisätään attribuuteiksi lisätietoja: linkin kohde ja se, miltä linkki näyttää kuvaruudulla.

Linkin kohde kertoo sen, minne selain siirtyy seuraavaksi linkkiä napsautettaessa. Kohteena voi olla toinen tiedosto tekemiesi sivujen joukossa, jokin osoite jossain päin Internetiä tai toinen paikka saman sivun sisällä.

Selaimessa linkit näkyvät kuvaruudulla muusta tekstistä poikkeavalla tyylillä esitettyinä, yleensä erivärisinä ja alleviivattuina. Tätä tekstiä nimitetään "kuumaksi tekstiksi", tai "kuumaksi sanaksi". Linkin voi valita viemällä hiiren osoittimen kuuman sanan päälle ja painamalla hiiren näppäintä (tai tekstipohjaisissa selaimissa nuolinäppäimien ja Return-näppäimen avulla). Kun linkki on valittu, selain hakee ja tuo näytölle näkyviin linkissä määritellyn tiedoston.

Linkki tehdään komennolla ****. Komennon jälkeen tuleva teksti on linkin kuuma teksti. Linkin määrittely loppuu komentoon ****. Lainausmerkkien välissä oleva teksti kertoo linkin osoittaman sivun sijainnin. Voit sijoittaa lainausmerkkien väliin tavallisen tiedostonnimen, jolloin tiedosto haetaan samasta hakemistosta kuin kuvaruudulla näkyvä sivu on haettu.

Aiemmin teit itsellesi kaksi HTML-sivua - **kotisivu.htm** ja **osoite.htm**. Nyt kokeillaan linkittää ne toisiinsa. Linkki tehtiin siis komennolla ****. Tämän komennon jälkeen kirjoitettiin linkin kuuma teksti eli se teksti, joka linkistä näkyy lukijalle. Kuuma teksti lopetetaan komentoon ****. Kun teet linkkiä itse tekemiesi sivujen välillä, voit laittaa osoitteeksi tiedoston nimen. Jos siis teet tiedostoon **kotisivu.htm** linkin,

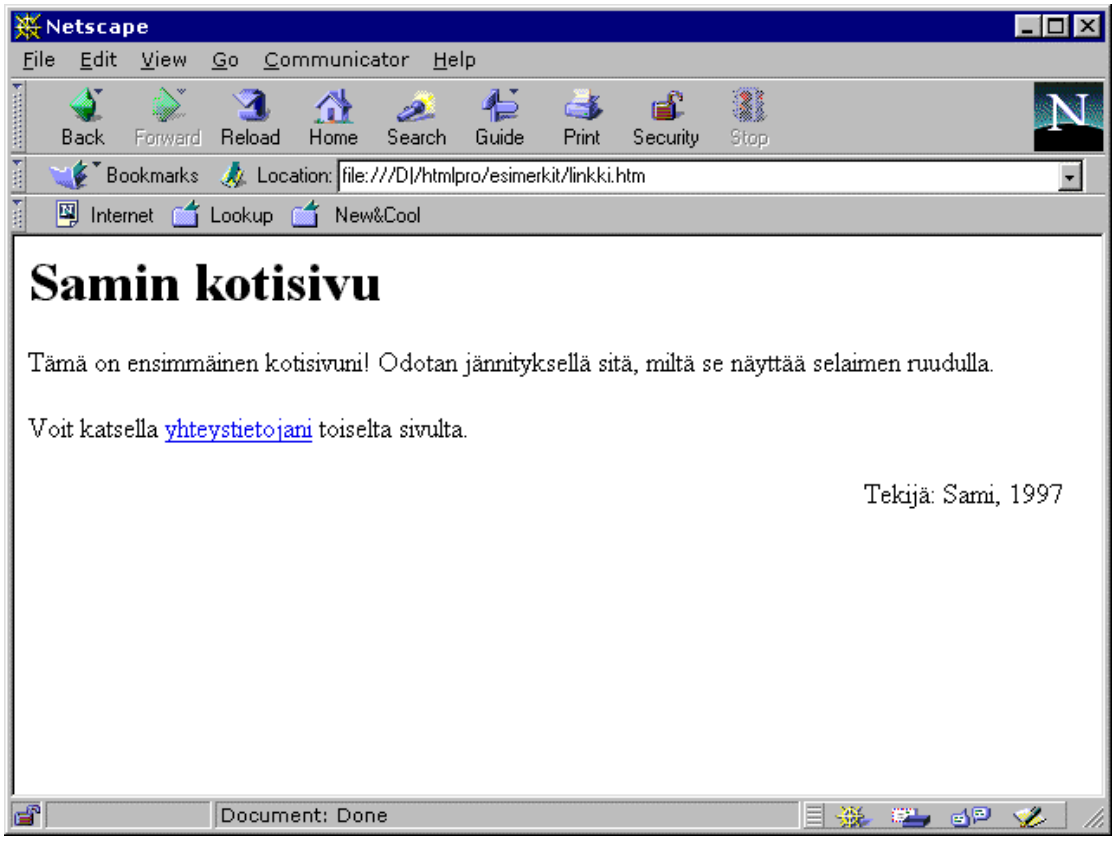
joka osoittaa toiseen tekemääsi tiedostoon, **osoite.htm**, linkkikomento näyttää seuraavanlaiselta:

```
<a href="osoite.htm">Kuuma teksti</a>
```

Ota kotisivusi esille HTML-editoriin, ja kokeile tehdä sinne linkki. Tulos voisi näyttää vaikkapa tällaiselta:

```
<h1>Samin kotisivu</h1>
<p>Tämä on ensimmäinen kotisivuni! Odotan
jännityksellä sitä, miltä se näyttää selaimen
ruudulla.</p>
<p>Voit katsella
<a href="osoite.htm">yhteystietojani</a>
toiselta sivulta.
<p align=right>Tekijä: Sami, 1997</p>
```

Kun kokeilet katsella sivuasi kuvaruudulla, huomaat kuinka olet tehnyt ensimmäisen linkkisi:



Kokeile napsauttaa linkkiäsi ja varmistaa sen toiminta. Nyt voit tehdä samalla tavalla itsellesi linkin tiedostosta **osoite.htm** takaisin tiedostoon **kotisivu.htm**. Voit myös tehdä useita uusia sivuja ja linkittää ne toisiinsa. Tällä tavalla voit tehdä nopeasti itsellesi kokonaisen sivukokonaisuuden!

LINKIT VERKKOON

Edellisessä esimerkissä kerrottiin, miten luodaan linkkejä saman hakemiston sisällä oleviin tiedostoihin. WWW tarjoaa myös mahdollisuuden tehdä linkkejä mihin tahansa WWW:ssä sijaitsevaan sivuun.

Muualle tehtävä linkki tehdään samalla tavalla kuin tiedostoon osoittava linkki. Tällä kertaa tiedoston nimen paikalle laitetaankin sivun verkko-osoite, URL. Tällä tavalla voit tehdä linkin, joka osoittaa vaikka toisella puolella maapalloa olevaan sivuun.

Otetaan jälleen esimerkki: olet rakentanut WWW-sivun, joka kertoo elokuvan historiasta. Sivun loppuun haluat liittää linkkejä mielenkiintoisille WWW-sivuille, jotka käsittelevät samaa asiaa. Oletetaan, että olet aiemmin WWW:ssä kulkiessasi useita sopivia sivuja ja kirjoittanut ylös niiden osoitteet.

Löytämiäsi osoitteita voisivat olla esimerkiksi

The Internet Movie Database (**http://www.imdb.com**)
sekä **Hollywood Online** (**http://www.hollywood.com**).

Nyt voit lisätä linkit sivun loppuun edellisestä luvusta tutulla tavalla. Tällä kertaa tiedoston nimen tilalle kirjoitetaan sivun URL.

```
<h2>Lisätietoja</h2>
<p>Jos kiinnostuksesi elokuviin heräsi, saat
lisätietoja elokuvista mm.
<a href= "http://www.imdb.com/">The Internet Movie
Database -tietokannasta</a> ja
<a href= "http://www.hollywood.com/">Hollywood
Online</a> -arkistosta.</p>
```

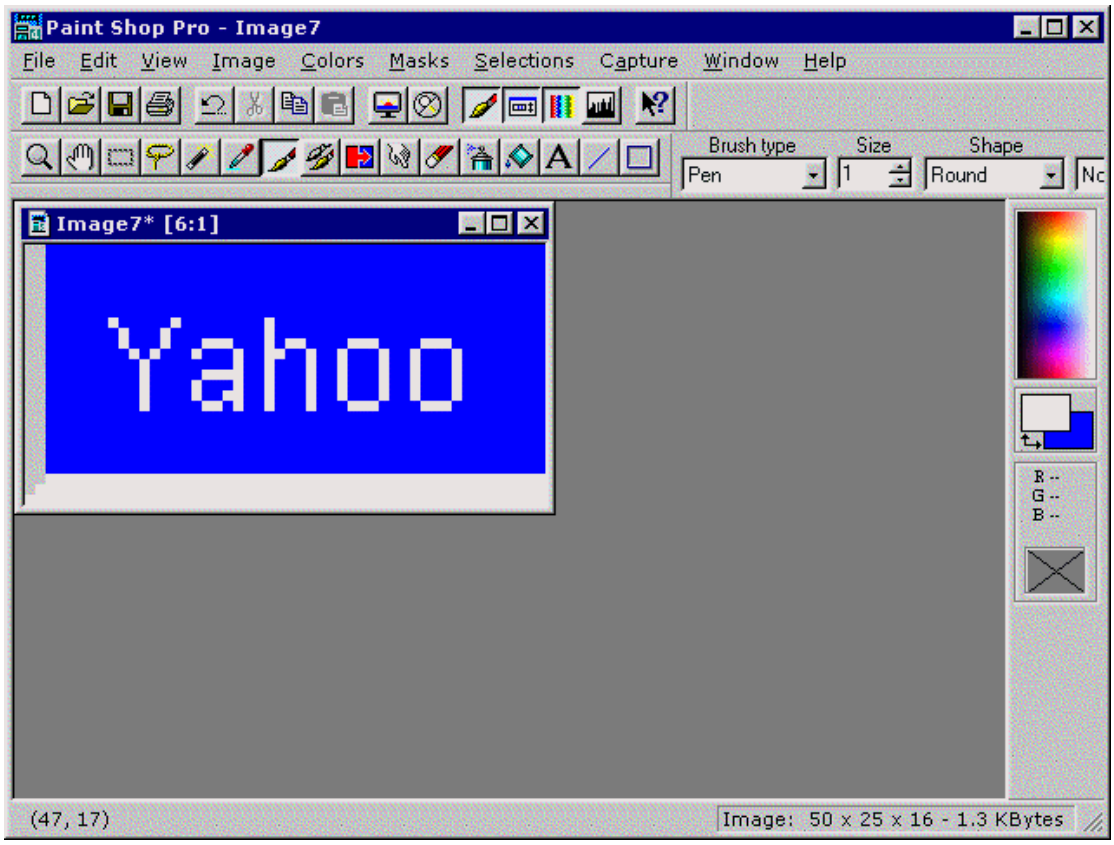
Kokeile tehdä kotisivullesi uusi alisivu (uuteen tiedostoon), jonne keräät oman linkkikokoelmasi. Anna tiedostolle nimeksi vaikkapa **linkit.htm** ja

tee kotisivultasi linkki tähän tiedostoon. Voit keräillä tälle sivulle omia suosikkilinkkejäsi, jotka haluat jakaa muiden lukijoiden kanssa.

KUVALINKIT

Linkin kuuma teksti voi olla myös kuva. Tällä tavalla voit tehdä sivuillesi kuvia, joita napsauttamalla käyttäjä pääsee siirtymään sivuilta toisille. Monilla sivuilla on pieniä painikkeen näköisiä kuvia, joista on tehty linkkejä.

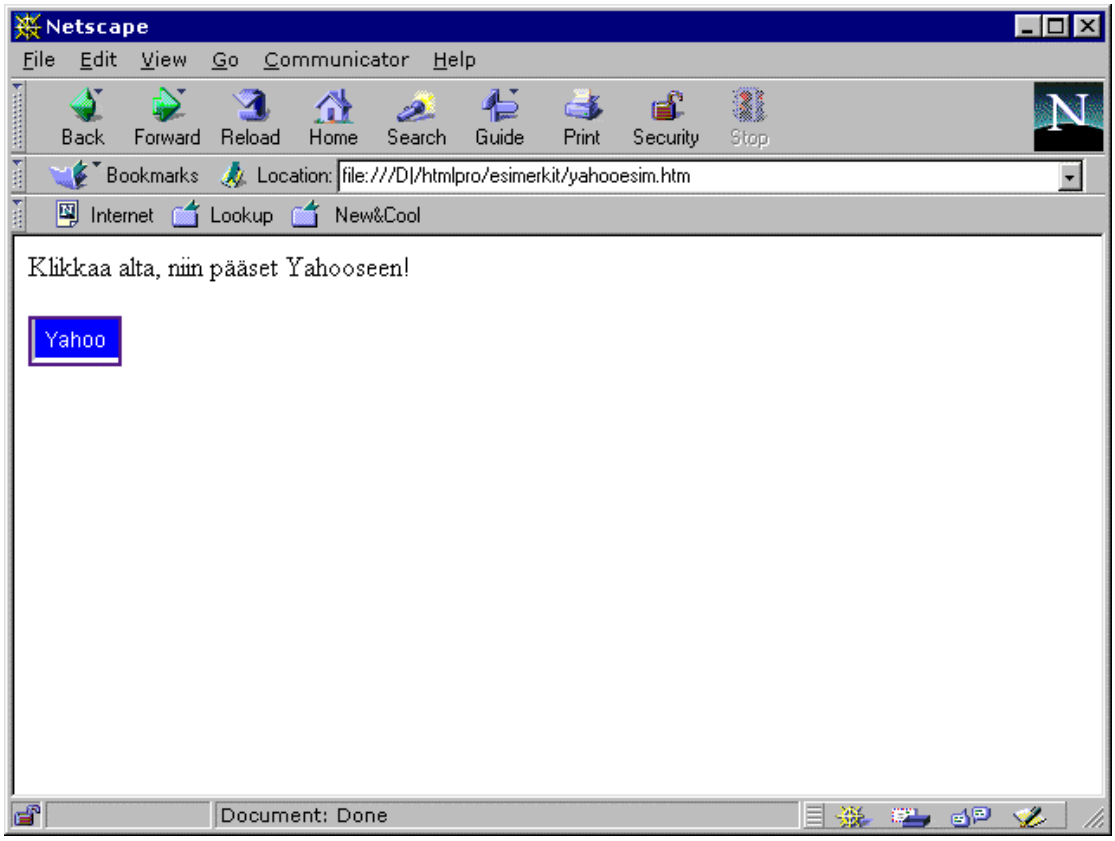
Voit tehdä kuvasta linkin samalla tavalla kuin teet tekstilinkinkin. Sijoita vain kuuman tekstin paikalle kuvakomento. Otetaan esimerkiksi pieni **gif**-kuva, josta on tehty painonapin näköinen. Piirretään ensin kuva kuvankäsittelyohjelmassa:



Sitten tallennetaan se samaan hakemistoon HTML-tiedoston kanssa nimellä "**yahoonappi.gif**" ja tehdään siitä painike, jonka napsautus vie **Yahoo**n sivuille:

```
<p>Klikkaa alta, niin pääset Yahooseen!
<p><a href="http://www.yahoo.com/"><img
src="yahoonappi.gif"></a>
```

Selaimessa näet painikkeesi toiminnassa:



Normaalisti selain tekee linkkinä toimivan kuvan ympärille reunuksen. Tämä reunus on samanvärinen kuin muiden linkkien kuuma teksti. Jos haluat poistaa reunuksen, voit lisätä linkin sisällä olevaan kuvakomentoon attribuutin **`border=0`**:

```
<p>Klikkaa alta, niin pääset Yahoo:hin!
<p><a href="http://www.yahoo.com/"><img
src="yahoonappi.gif" border=0></a>
```

Tee tällöin kuvasta sellainen, että sivusi lukija ymmärtää kuvan olevan linkki (jos haluat, että käyttäjä hoksaa napsauttaa linkkiä).

NÄKYMÄTTÖMIÄ MUTTA TÄRKEITÄ

WWW-sivuilla käytetään myös komentoja, jotka eivät näy käyttäjän ruudulla. Nämä komennot kuvaavat yleensä jollain tavalla sivun rakennetta tai toimintaa. Alla näet esittelyn näkymättömistä, mutta tärkeistä komennoista.

KOMMENTOINTI

Voit lisätä tiedostoosi kommentteja muistutukseksi itsellesi tai muille HTML-tiedostoa tutkiville. Kommentit eivät tule näkyviin selaimen ruudulle vaan ne näkyvät ainoastaan esimerkiksi katsottaessa sivua selaimen **View Source** -toiminnolla. Kommenttien avulla voit vaikkapa selvittää jotain tekemääsi teknistä ratkaisua tai merkitä itsellesi kohtia, joita haluat muokata myöhemmin.

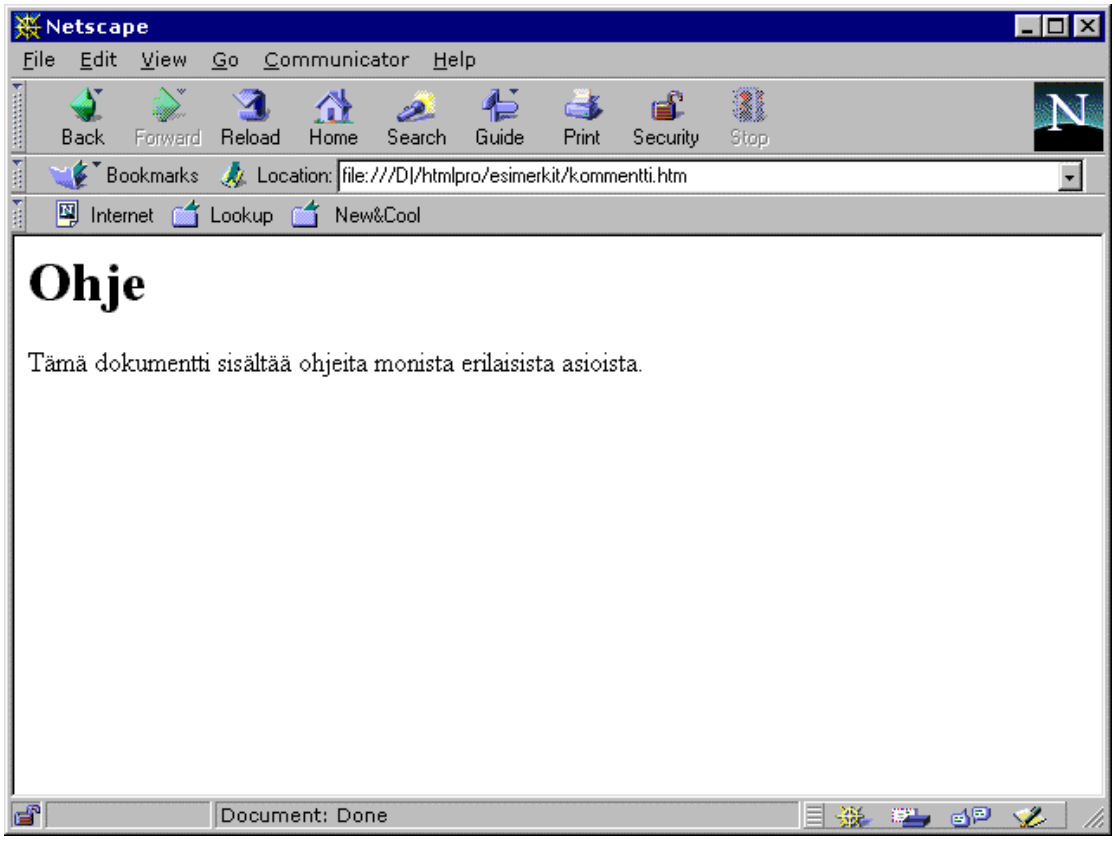
Kommenttien avulla voit myös merkitä osia tiedostosta niin, etteivät ne näy lopullisella sivulla, kuitenkaan poistamatta varsinaisia komentoja. Lisäksi kommentit saattavat sisältää komentoja erilaisille editoriohjelmille ja palvelimille.

Kommentti aloitetaan merkeillä **<!--** ja lopetetaan merkkeihin **-->**. Kaikki näiden merkkien väliin jäävä teksti on kommenttia. HTML:n kommentteja ei voi sijoitella sisäkkäin.

Esimerkkejä:

```
<h1>Ohje</h1>
<!-- Ylläoleva teksti on otsikko –->
<p>Tämä sivu sisältää ohjeita monista erilaisista
asioista.</p>
```

Kuvaruudulle tulevassa sivussa kommentti ei ole näkyvissä:



Jos haluat tehdä useamman kuin yhden rivin mittaisen kommentin, sijoita kommenttimerkit jokaiselle riville:

```
<h1>Puhelinvastaajan toiminta</h1>
<!-- Tällä sivulla selitetään -->
<!-- puhelinvastaajan toimintaa.   -->
```

Jos haluat poistaa väliaikaisesti osan sivusta, voit ympäröidä sen kommenteilla:

```
<p>Tervetuloa kotisivulleni!
<!-- <p>Tämä sivu on rakenteilla! -->
```

SIVUN RAKENNE

Edellä esitetyt komennot vaikuttavat suoraan sivun lukijan näytössä näkemiin teksteihin. HTML sisältää lisäksi useita sivun rakennetta kuvailevia komentoja.

Sivun rakennetta kuvaavat komennot antavat vinkkejä selaimille ja erilaisille ohjelmille. Kun esimerkiksi sivulle on määritelty otsikkotiedot, ohjelmat osaavat ottaa sivusta erilleen sivua kuvaavat tiedot helposti ja nopeasti selaamatta koko tiedostoa. Erilaiset hakutietokannat pystyvät nopeasti lisäämään sivulta tarvittavat tunnistetiedot tietokantaansa.

ENSIMMÄINEN JA VIIMEINEN KOMENTO

Komennot **<html>** ja **</html>** ympäröivät koko HTML-sivua. Normaalisti sivun alkuun sijoitetaan komento **<html>** ja sivun viimeiseksi komennoksi sijoitetaan lopetuskomento **</html>**. Selain käsittää kaiken komentojen väliin jäävän tekstin olevan HTML-kieltä, jossa olevat komennot täytyy ottaa huomioon.

Kuten olet aiemmin huomannut, sivut toimivat nykyisissä selaimissa ilman näitä komentojakin, mutta niiden käyttäminen kannattaa ottaa tavaksi - joskus saatat törmätä selaimeen, joka ei osaa näyttää sivuasi kunnolla ellei se sisällä näitä komentoja.

<html>-komentojen väliin sijoitetaan sivun otsikkotiedot (**<head>** ja **</head>**-komentojen väliin) ja runko **(<body>** ja **</body>**-komentojen väliin) - näistä komennoista lisää seuraavassa.

OTSIKKOTIEDOT

Itse HTML-tiedosto jaetaan otsikkotietoihin ja varsinaiseen tekstin runkoon. Otsikkotiedot sisältävät sivun rakennetta ja muihin sivuihin liittyviä yhteyksiä kuvaavia komentoja. Otsikkoalueella sijaitsevat tiedot eivät tule yleensä näkyviin selaimessa (poikkeuksena on sivun nimi, jonka selain näyttää sille varatussa paikassa).

Otsikkotiedot aloitetaan komennolla **<head>** ja lopetetaan komentoon **</head>**. Otsikkotietojen sisään lisätään sivun nimi ja muita sivua kuvaavia komentoja.

SIVUN NIMI

Sivulle annetaan nimi komennolla **title**. Tämä on yksi tärkeimpiä sivun otsikkotietoihin liitettävistä komennoista. Jokaisen HTML-sivun tulisi sisältää määritelmä sivun nimestä (**Title**). Sivun nimi kuvaa yksiselitteisesti

sivun sisältöä. Huomaa, että tällä komennolla tehty sivun nimi ei tarkoita HTML-tiedoston nimeä!

Kun sivua luetaan, sen nimi tulee yleensä näkyviin selaimen ikkunan yläreunaan tai selaimessa sille varatulle alueelle. Netscape ja Internet Explorer näyttävät sivun nimen otsikkopalkissa. Lisäksi sivun nimeä käytetään selaimen historia- ja kirjanmerkkivalikoissa.

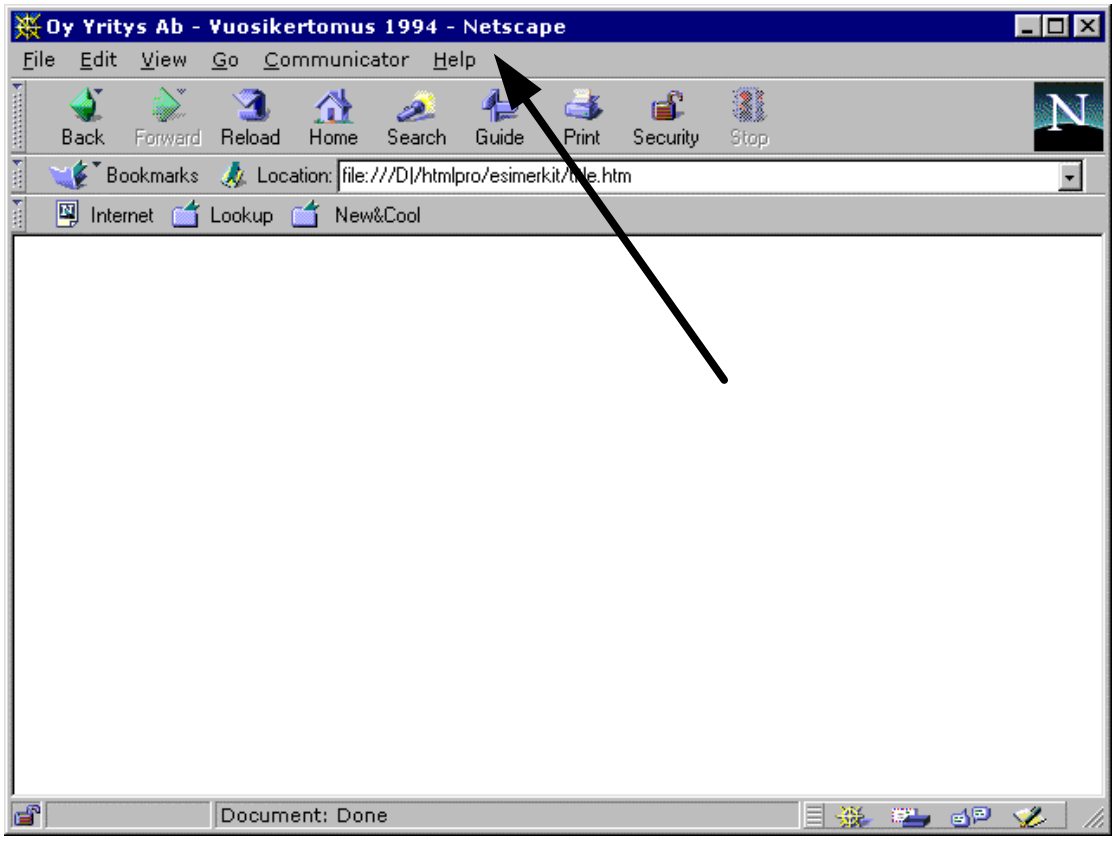
Useat WWW:n sisältöä indeksoivat hakuohjelmat tallentavat sivusta sen nimen omaan tietokantaansa. Tällä tavalla kyseisen hakuohjelman käyttäjät voivat etsiä haluamiaan tietoja sivujen nimien perusteella.

Sivun nimeksi kannattaa keksiä jokin sivua hyvin kuvaava teksti. Jos sivulla on esimerkiksi yrityksen esittely, sen nimeksi ei kannata laittaa "Esittely", sillä maailmalla on varmasti useita muitakin sivuja samalla nimellä. Parempi nimi voisi olla vaikkapa "Oy Yritys Ab — Esittely". Nimen pituudella ei ole määriteltyä rajaa, mutta käytännössä on varminta pitää sivujen nimet alle 64 merkin mittaisina (tällaiset nimet sopivat parhaiten selaimen otsikkopalkkeihin ja valikoihin).

Esimerkki sivun nimen asettamisesta:

```
<head>
<title>Oy Yritys Ab – Vuosikertomus 1994</title>
</head>
```

Näet sivun nimen selaimen otsikkopalkissa:



RUNKO

WWW-sivun runko määritellään **body**-komennolla. Rungon sisään sijoitetaan varsinainen HTML-sivu. Kaikki rungon sisällä oleva teksti näkyy käyttäjälle sivuna otsikkoineen ja leipäteksteineen. Rungon sisällä voit käyttää kaikkia niitä komentoja, joita ei käytetä otsikkotiedoissa.

Esimerkki:

```
<body>
<h1>Oy Yritys Ab</h1>
<p>Oy Yritys Ab perustettiin vuonna 1989. Sen toiminta
keskittyy pääasiassa hypermediaan.</p>
</body>
```

Body-komentoon liitetään myös sivun väreihin ja taustakuviin liittyviä attribuutteja (näistä lisää myöhemmin).

KOKONAINEN HTML-TIEDOSTO

Kokonaiseen HTML-tiedostoon kuuluvat siis edellä esitellyt osat: HTML-sivun määritelmä, otsikkotiedot ja runko. Otsikkotiedoissa sijaitsee yleensä ainakin sivun nimi. Seuraavassa on esimerkki HTML-sivun valmiista pohjasta:

```
<html>
<head>
<title> Sivun nimi </title>
</head>
<body>
<!-- Varsinainen sivu tulee tähän. -->
</body>
</html>
```

Yllä oleva pohja on itsessään jo valmis HTML-tiedosto. Voit käyttää sitä pohjana, jonka päälle voit rakentaa uusia sivuja. Kopioi teksti koneesi kovalevylle, esimerkiksi tiedostoon nimeltään **pohja.htm**. Aina kun alat kirjoittaa uutta sivua, voit ladata tuon tiedoston, muuttaa sivun nimen ja kirjoittaa sivun tekstin **body**-komentojen väliin. Esimerkkitiedostossa **body**-komentojen välissä on kommenttirivi, joka näyttää varsinaisen tekstin paikan. Voit poistaa tuon rivin lopullisesta versiosta. Monet HTML-editorit osaavat tehdä nämä komennot automaattisesti, kun aloitat uuden sivun tekemisen.

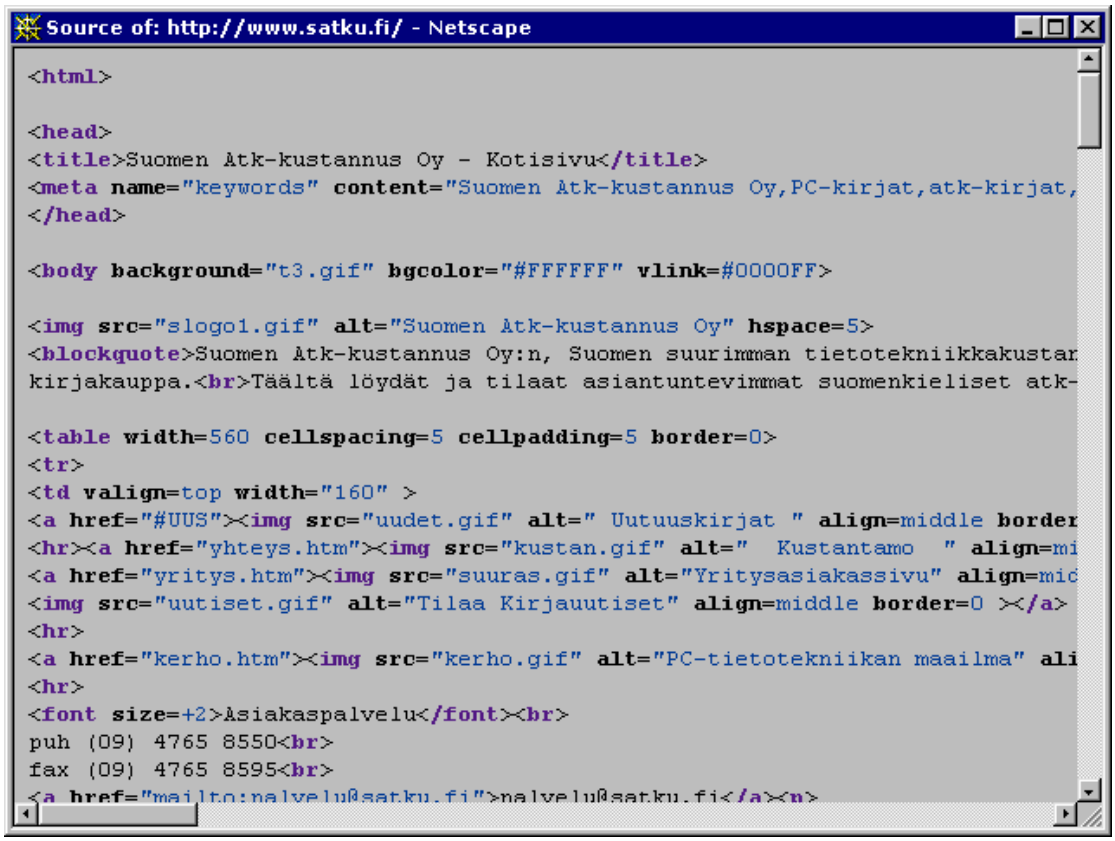
Kokeile lisätä aiemmin tekemiisi sivuihin alkuun ja loppuun tarvittavat komennot. Anna myös jokaiselle tekemällesi sivulle kuvaava nimi.

MALLIA MUIDEN SIVUISTA

Nyt kun HTML:n perustoiminnot ovat sinulle tuttuja, osaat varmasti katsella Internetissä sijaitsevia WWW-sivuja kokonaan uudella tavalla. Osaat tehdä monia sivuilla näkyviä asioita ja törmäät tuttuihin toteutustapoihin.

Yksi WWW:n erikoispiirteistä on se, että jokaisen verkossa sijaitsevan WWW-sivun HTML-koodi on katseltavissa. Voit siis katsella, miten muiden tekemät WWW-sivut on tehty ja tutkia miten ne on tehty.

Minkä tahansa WWW-sivun tekemiseen käytetty HTML-koodi saadaan näkyviin selaimen **View Source** -toiminnolla. Netscapessa ja Internet Explorerissa tämä toiminto löytyy **View**-valikosta. Kokeile etsiä selaimeesi jokin WWW-sivu ja katsella sen toteutusta selaimellasi!



Saat sivun HTML-koodin näkyviin myös napsauttamalla hiiren oikeanpuoleista nappia sivun päällä ja valitsemalla esiin tulevasta valikosta toiminnon **View Source**. Tällä tavalla pystyt tutkimaan sivuja, jotka on jaettu useampiin ali-ikkunoihin.

Verkosta löytyvät sivut sisältävät todennäköisesti komentoja, joiden kanssa et ole vielä tuttu. Lue kuitenkin kirjaa eteenpäin, niin opit kaikki sivunsuunnittelun niksit!

Muista myös, että koska sivun koodin katselua ei voi estää millään tavalla, lukijasi voivat katsella myös sinun tekemiesi sivujen koodia.

KOTISIVU VERKKOON - SIVUN PÄIVITTÄMINEN PALVELIMELLE

Tähän asti olet muokannut kotisivujasi oman tietokoneesi kovalevyllä. Kun kotisivusi on siinä kunnossa, että haluat julkistaa sen koko maailman luettavaksi, voit kopioida sen oman Internet-operaattorisi palvelinkoneelle.

Operaattorin palvelinkoneella ollessaan sivut näkyvät osana Internet-palveluntarjoajasi sivuja. Niiden osoitteen alussa on yleensä operaattorisi osoite. Osoitteessa on lisäksi oma käyttäjätunnuksesi, joten oman kotisivusi osoite näyttää yleensä seuraavantapaiselta:

```
http://operaattori.fi/~tunnuksesi/
```

Jos sinulla ei ole vielä hankittuna Internet-yhteyttä, joudut ensimmäiseksi ostamaan sellaisen. Hankkiessasi yhteyttä voit kysyä myös lisätietoja operaattorin tarjoamista kotisivumahdollisuuksista: onko kotisivun tekeminen ylipäätään mahdollista, voitko päivittää omaa kotisivuasi omalta koneeltasi ja miten laajan kotisivun voit tehdä.

Sivujen siirtämistä varten tarvitset jonkin verran tietoa operaattoriltasi. Miltei kaikilla operaattoreilla on tarjolla opas sivujen siirtoa varten, josta voit katsoa tarvitsemasi tiedot - useimmissa oppaissa kerrotaan myös yksityiskohtaisesti, miten siirto tapahtuu. Alta löydät yleiskäyttöiset ohjeet, jotka toimivat suurimmalla osalla operaattoreista. Linkkejä operaattoreiden ohjesivuille löydät alempaa.

Kotisivujen siirtoon tarvitset seuraavat tiedot:

- Osoite, johon valmiit sivut siirretään. Tämä on tietokoneen nimi, joka sisältää hakemistotilaa käyttäjien kotisivuille, esimerkiksi **Kolumbus**-käyttäjillä **`koti.kolumbus.fi`**, **PersonalEunet**-käyttäjillä **`personal.eunet.fi`** tai **Clinetin** käyttäjillä **`ftp.clinet.fi`**.

- Käyttäjätunnus ja salasana, jolla voit avata yhteyden. Nämä ovat yleensä samat, joita käytät normaalistikin Internet-yhteytesi avaamiseen.

- Hakemisto, jonne sivut siirretään. Tämä saattaa olla esimerkiksi nimeltään **`public_html`** tai **`www`**.

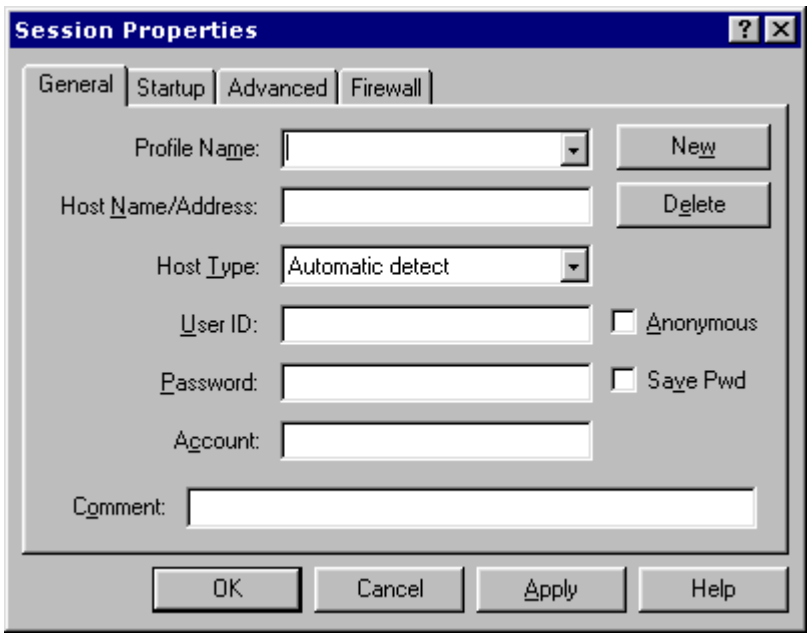
FTP:N KÄYTTÖ

Tarvitset valmiiden sivujen julkistamiseen tiedostonsiirto-ohjelman. Jotkut HTML-editorit sisältävät tällaisen ohjelman sisäänrakennettuna (lue lisätietoja kyseisen ohjelman opasmateriaalista), mutta voit aina käyttää siirtämiseen myös erillistä siirto-ohjelmaa.

Yksi eniten käytettyjä tiedostonsiirto-ohjelmia on Windows-koneissa toimiva **WS-FTP**. Se on julkisohjelma, eli voit kopioida sen maksutta käyttöösi verkossa sijaitsevista julkisohjelmapalvelimista. Macintosh-koneilla voit käyttää esimerkiksi ohjelmaa nimeltään **Fetch**, jonka toiminnot vastaavat **WS-FTP**:n toimintoja.

ALKUASETUKSET

Kun käynnistät **WS-FTP**:n, saat ensimmäisenä näytöllesi ikkunan, jossa sinulta kysytään tarkempia tietoja operaattorisi palvelimesta:



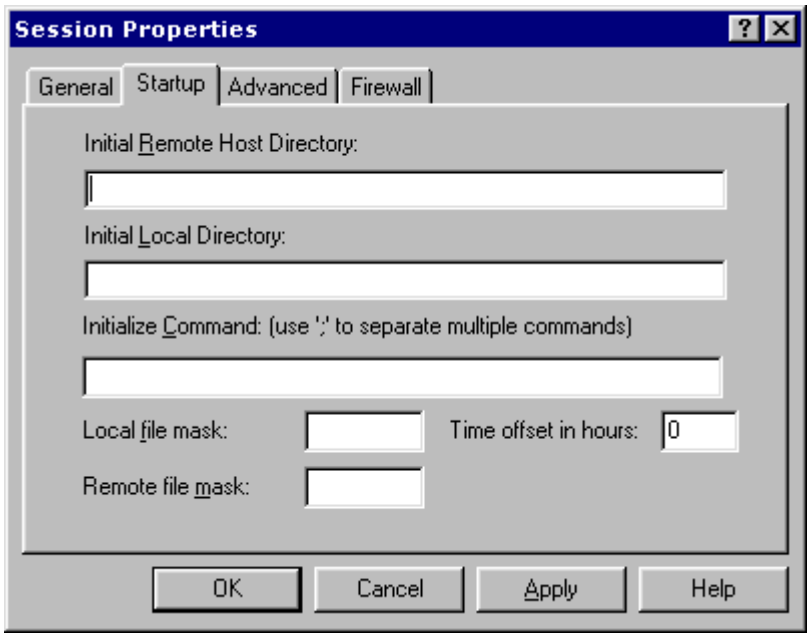
Jos ruudullasi näkyvä lomake ei ole tyhjä, kuten yllä näkyvässä kuvassa, napsauta kerran painiketta **New**. Tämän jälkeen voit alkaa täyttää tietojasi lomakkeeseen.

Täytä operaattorisi tietokoneen osoite kohtaan **Host Name/Address**. Tämä nimi on usein sama kuin sen koneen nimi, jossa operaattorisi kotisivu sijaitsee. Tässä esimerkissä siirrämme sivut Tampereen yliopiston palvelimeen, joten koneen nimi on silloin **`www.uta.fi`**.

Tarvitset sivujen päivitykseen aina käyttäjätunnuksen ja salasanan. Nämä ovat usein samat, joita käytät avatessasi Internet-yhteyttä koneeltasi. Täytä tunnuksesi kohtaan **User ID**. Salasanasi voit kirjoittaa kohtaan **Password**. Voit jättää salasanan vielä tässä vaiheessa tyhjäksi, jolloin **WS-FTP** kysyy sitä sinulta tarvittaessa.

Jos kirjoitat salasanasi jo tässä vaiheessa ja rastitat ruudun **Save Pwd**, **WS-FTP** muistaa salasanasi myöhemminkin eikä sinun tarvitse kirjoittaa sitä joka päivityksen yhteydessä uudelleen. Huomaa kuitenkin, että salasanasi tallentuu silloin tietokoneesi kovalevylle ja kuka tahansa, joka pääsee käyttämään konettasi pystyy näkemään salasanasi ja muokkaamaan operaattorin palvelimella sijaitsevia sivuja! Älä käytä tätä toimintoa, jos päivität sivujasi julkisella tietokoneella (kirjastossa, nettikahvilassa tai oppilaitoksen mikroluokassa).

Vaihda seuraavaksi lomakkeen yläreunassa näkyvistä painikkeista näkyviin välilehti **Startup** (älä paina vielä **OK**-painiketta!). Saat näkyviin seuraavanlaisen lomakkeen:

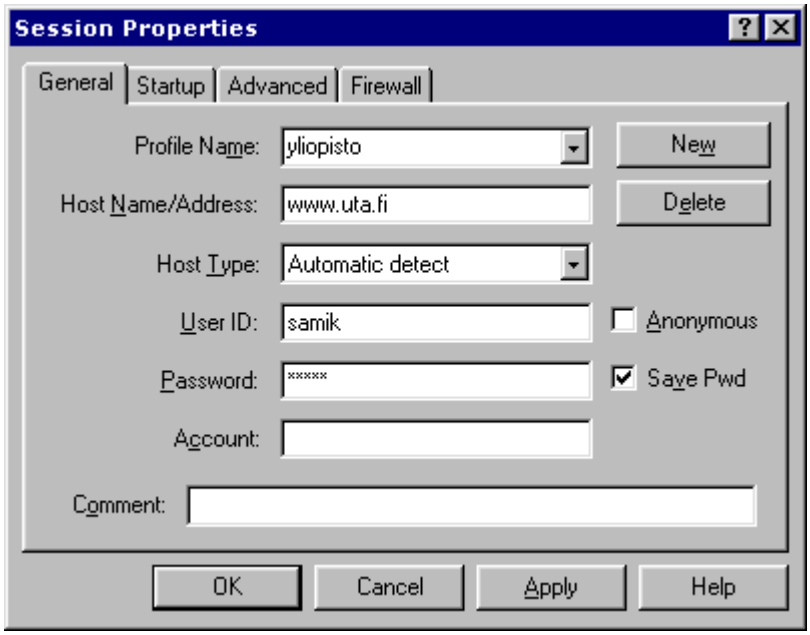


Kohtaan **Initial Remote Host Directory** kirjoitetaan sen hakemiston nimi, jonne sivut siirretään palvelinkoneella. Yleensä tämän hakemiston nimi on **public_html** tai **www**, tai sitten se voi olla myös tyhjä.

Seuraavaan kohtaan, **Initial Local Directory**, voit kirjoittaa omalta kovalevyltäsi sen hakemiston nimen, jonne olet tallentanut tekemiäsi WWW-sivuja. Jos olet esimerkiksi tallentanut sivusi C-levyasemalle hakemistoon nimeltään **omat**, kirjoita lomakkeeseen **C:\omat**.

Lopuksi voit vielä tallentaa kaikki kirjoittamasi asetukset **WS-FTP**:n muistiin, niin että sinun ei tarvitse kirjoittaa samoja tietoja uudelleen seuraavalla kerralla tiedostoja siirtäessäsi. Siirry takaisin **General**-välilehteen napsauttamalla sen nimeä lomakkeen yläreunasta. Keksi antamillesi tiedoille jokin yksisanainen nimi, joka on helppo muistaa ja kirjoita se laatikkoon **Profile Name**.

Esimerkissä olen antanut Tampereen yliopiston palvelimelle tarvittavat asetukset ja tallettanut ne **WS-FTP**:n muistiin nimelle "yliopisto":

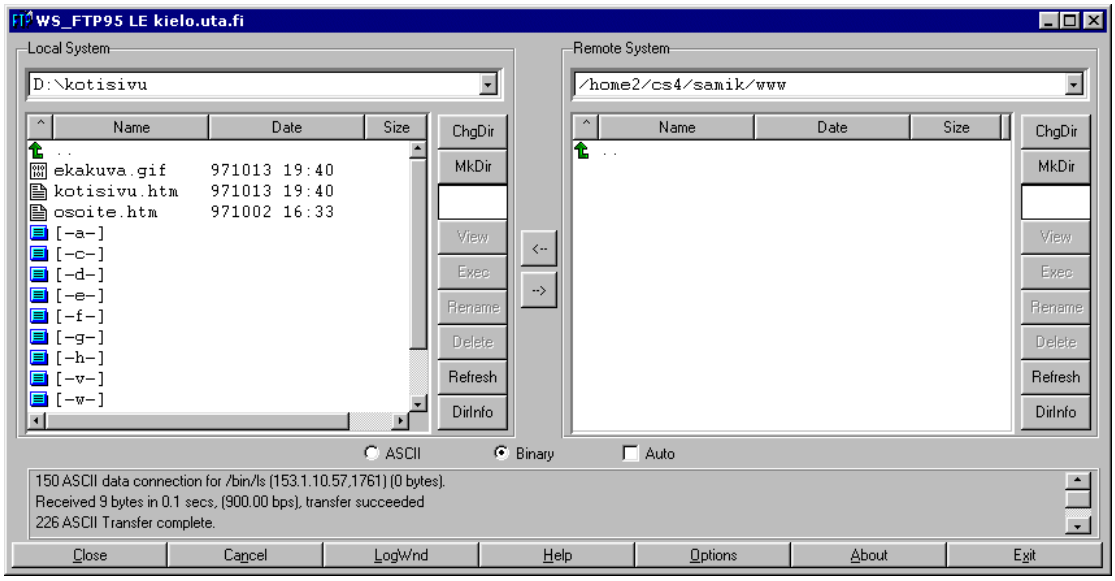


Tämän jälkeen **WS-FTP** muistaa kirjoittamasi asetukset ja voit käyttää niitä yhä uudelleen ja uudelleen käyttämällä pelkästään keksimääsi nimeä. Yllä olevan esimerkin jälkeen voin käynnistää **WS-FTP**:n, kirjoittaa kohtaan **Profile Name** antamani nimen "yliopisto", ja **WS-FTP** täyttää automaattisesti muut kohdat antamillani tiedoilla.

TIEDOSTOJEN SIIRTO

Kun painat **OK**-painiketta, **WS-FTP** avaa yhteyden operaattorisi palvelinkoneeseen ja siirtyy automaattisesti oikeisiin hakemistoihin.

Tämän jälkeen ohjelma on valmis ja odottaa komentojasi tiedostonsiirron aloittamiseksi.

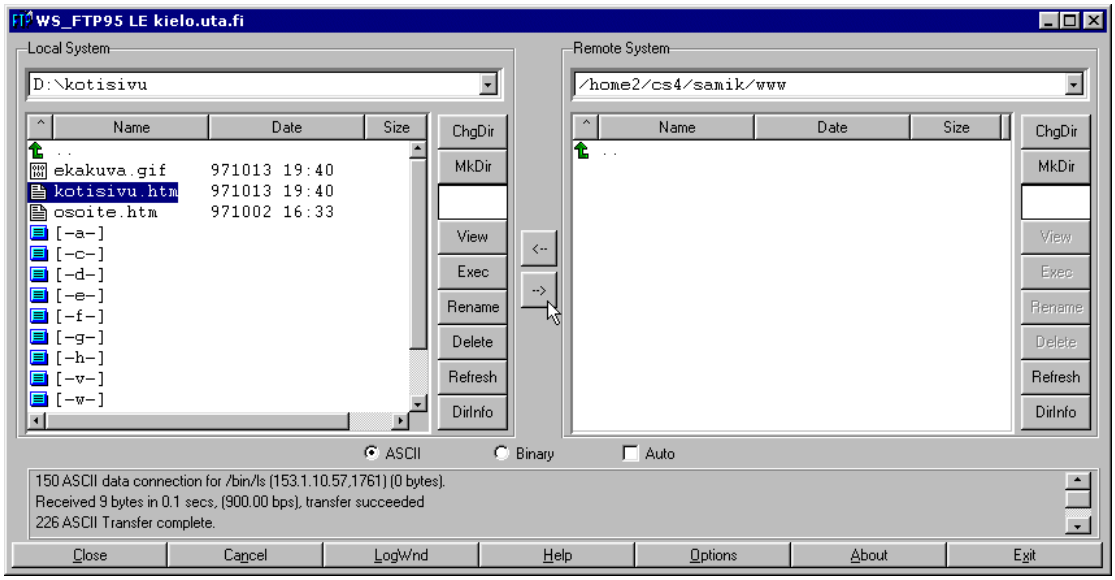


WS-FTP:n ruudulla on näkyvissä kaksi aluetta: vasemmalla puolella näet omassa tietokoneessasi sijaitsevat tiedostot ja oikealla puolella operaattorisi palvelimessa olevat, jo julkaistut tiedostot. Kun siirrät kotisivuasi palvelimelle ensimmäistä kertaa, oikea puoli näkyy tyhjänä (et ole julkistanut vielä mitään tietoja).

Näiden alueiden alapuolella näet painikkeet nimiltään **ASCII** ja **Binary**. Kun siirrät tiedostoja, sinun tulee aina tarkistaa tiedoston siirtomuoto. Pelkkää tekstiä sisältävät tiedostot (HTML-tiedostot ja muut tekstitiedostot) siirretään **ASCII**-muodossa ja kaikki muut tiedostot (kuvat, äänet, ohjelmat jne.) siirretään **Binary**-muodossa. Voit valita siirtomuodon napsauttamalla oikeaa painiketta.

Jos olet seurannut kirjan esimerkkejä, sinulla on siirrettävänä kolme tiedostoa: kaksi HTML-tiedostoa ja piirtämäsi kuva. Siirretään ensimmäisenä HTML-tiedostosi.

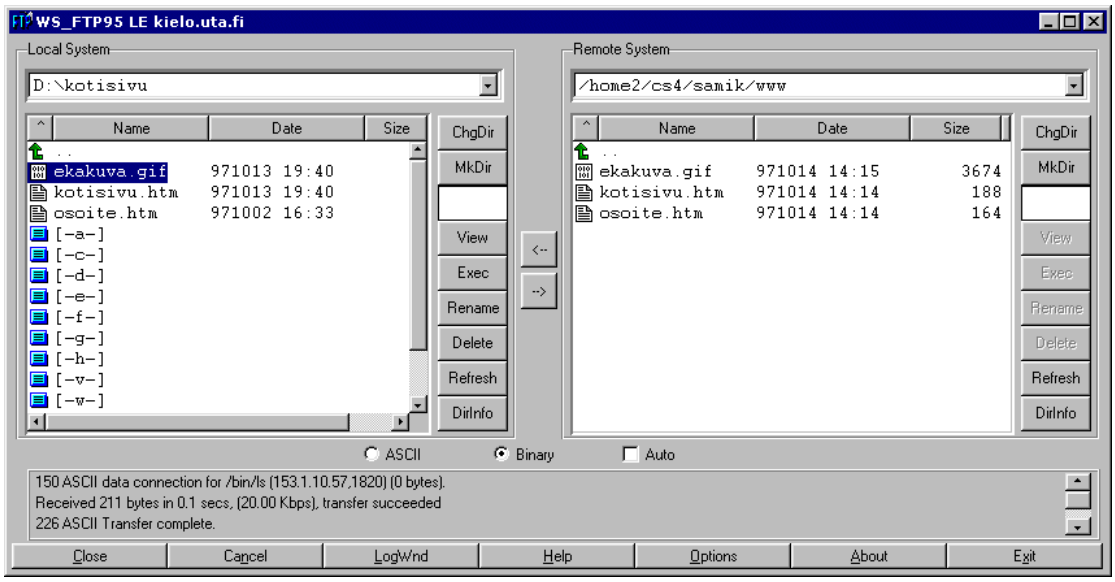
Kun olet siirtämässä tekstiä, napsauta hiirelläsi päälle ASCII-siirtomuoto valkoisten alueiden alla olevasta pyöreästä painikkeesta. Tämän jälkeen voit siirtää HTML-tiedostot napsauttamalla niiden nimeä vasemmalta puolelta ja sen jälkeen oikealle päin osoittavaa nuolta. Tämä kertoo **WS-FTP**:lle, että haluat siirtää tiedoston omalta koneeltasi operaattorin palvelimelle (vasemmalta oikealle):



Kun napsautat nuolta, **WS-FTP** siirtää tiedoston (näet muutaman sekunnin ajan ruudullasi siirto-ikkunan) ja antaa lopuksi äänimerkin onnistuneesta siirrosta. Siirrä myös toinen HTML-tiedostosi samalla tavalla.

Kun olet siirtänyt HTML-tiedostot, voit siirtää vielä kuvasi. Vaihda ensin siirtomuodoksi **Binary** napsauttamalla samannimistä painiketta, napsauta sitten kerran kuvatiedostoa vasemmalta ja oikealle päin osoittavaa nuolta.

Lopuksi kaikki tiedostot ovat kopioituna operaattorisi palvelimelle:



Tämän jälkeen kotisivusi on julkaistu verkossa! Voit katsella sitä selaimellasi syöttämällä sisään kotisivusi URL-osoitteen.

KOTISIVUSI OSOITE

Kotisivusi osoitteen saat selville operaattorisi tarjoamista ohjeista. Yleensä osoite muodostuu siirrossa käyttämästäsi koneen nimestä ja käyttäjätunnuksestasi seuraavalla tavalla:

```
http://operaattorin.kone/~tunnuksesi/tiedosto
```

Operaattorin koneen nimen kohdalle tulee siis yleensä se nimi, jota käytit siirrossa (esimerkiksi **koti.kolumbus.fi**). Nimi saattaa olla myös operaattorin varsinaisen palvelinkoneen nimi, vaikkapa **www.clinet.fi**.

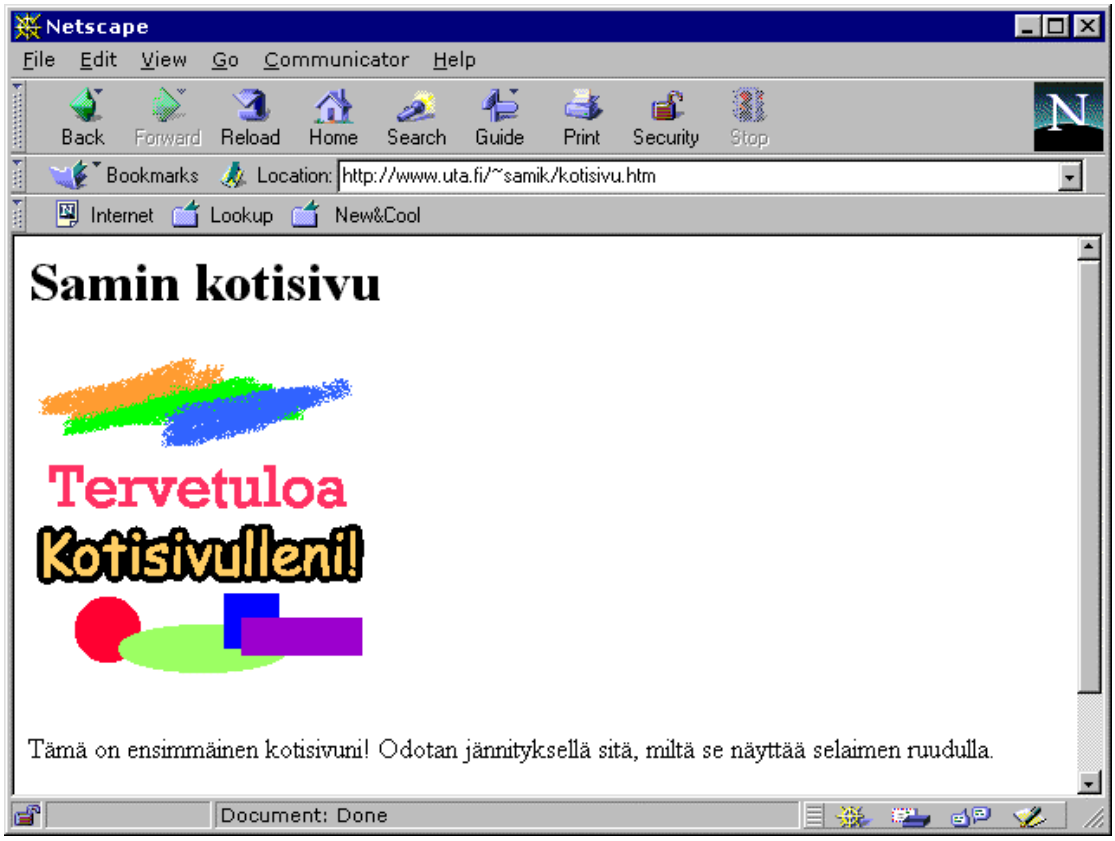
Matomerkin (**~**) jälkeen tulee tunnuksesi - tämä on yleensä se sama käyttäjätunnus, jota käytit tiedostojen siirrossa.

Viimeisen kauttaviivan jälkeen tulee vielä kotisivusi varsinainen tiedostonnimi - meidän esimerkissämme se oli **`kotisivu.htm`**.

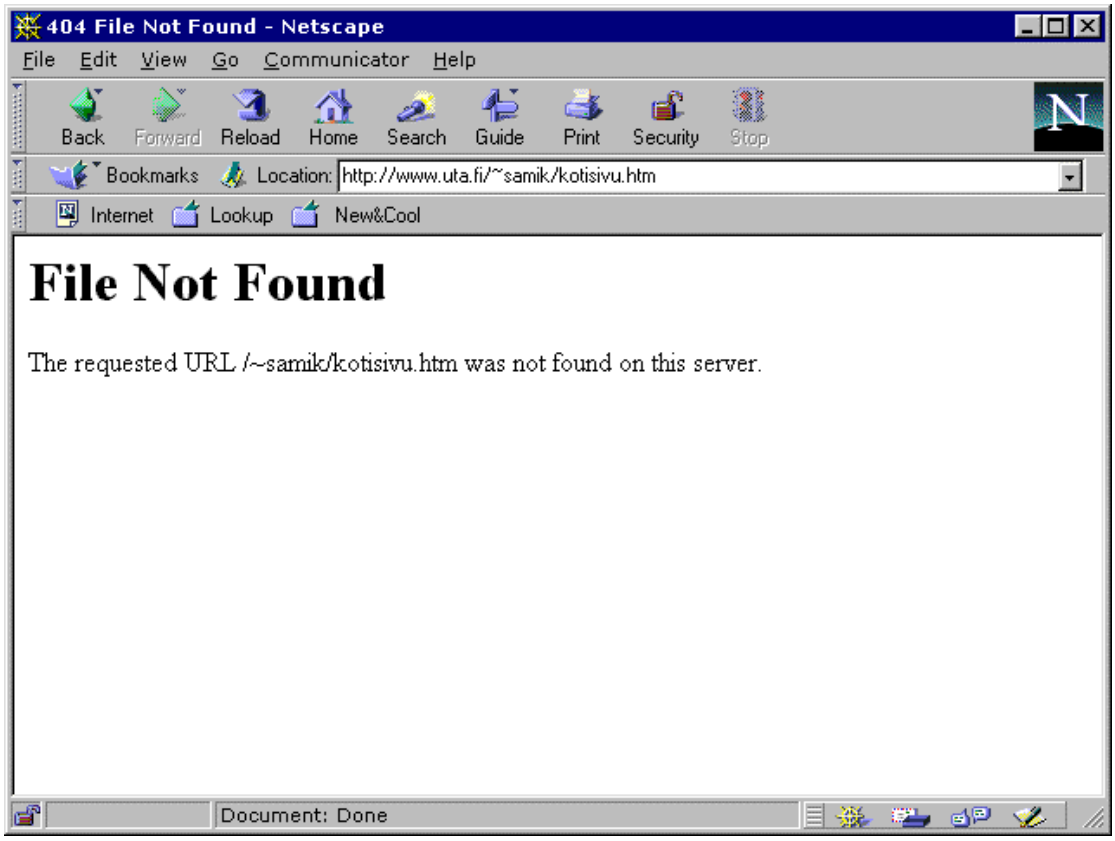
Yksi esimerkki oikeasta osoitteesta voisi olla siis vaikkapa:

```
http://www.uta.fi/~samik/kotisivu.htm
```

Kun syötät osoitteen selaimesi osoitekenttään, saat näkyviin kotisivusi:



Joskus saattaa kuitenkin käydä niin, että sivusi lataus ei onnistukaan. Useimmiten saat tällaisissa tilanteissa selaimesi kuvaruudulle virheilmoituksen:



Jos sivusi lataus ei onnistu, syy voi olla jokin seuraavista:

- Siirsit tiedostot väärään hakemistoon. Tarkista operaattorisi ohjeista tai heidän asiakaspalvelustaan oikean hakemiston nimi.

- Siirto ei onnistunut. Kokeile siirtää tiedostot uudelleen vanhojen päälle.

- Osoite on väärin. Tarkista operaattorisi ohjeista tai asiakaspalvelusta osoitteeseen tulevan palvelinkoneen nimi. Tarkista myös, kuuluuko osoitteeseen matomerkki (**~**), vai käytetäänkö operaattorisi palvelimessa jotain toista merkintätapaa.

- Tiedostojen nimet ovat väärin. Tarkista, että olet käyttänyt tiedostojen nimissä vain pieniä kirjaimia ja että et ole käyttänyt ääkkösiä tai väli-lyöntejä.

ETUSIVUTIEDOSTO

Aiemmissa esimerkeissä kotisivusi osoite koostui aina koneen nimestä, hakemistosta ja tiedoston nimestä. Osoitetta on mahdollista lyhentää siten, että siitä jätetään pois tiedoston nimi. Verkossa liikkuessasi olet varmasti

käyttänyt usein sivuja, joiden osoitteessa on vain tietokoneen tai tietokoneen ja hakemiston nimi:

```
http://www.cnn.com/
http://www.uta.fi/maps/
```

Kun WWW-palvelimelta haetaan osoitetta, jossa ei ole ilmoitettu tiedoston nimeä, se yrittää etsiä hakemistosta etusivutiedostoa. Useimmissa palvelimissa tämän tiedoston nimi on **index.htm** tai **index.html**.

Voit vaihtaa kotisivusi tiedoston nimeksi **index.html**, jolloin voit viitata omaan kotisivuusi käyttäen pelkästään tietokoneen ja hakemiston nimeä. Minun kotisivuni osoite on siis:

```
http://www.uta.fi/~samik/
```

Tämä osoite on sama kuin alla näkyvä osoite:

```
http://www.uta.fi/~samik/index.html
```

Jos etusivutiedosto ei toimi operaattorisi palvelimessa, on mahdollista että juuri siinä palvelimessa etusivutiedoston nimi on jokin toinen. Kokeile muuttaa sivusi nimeksi **home.html** tai **default.html** ja katsoa jos jompikumpi näistä toimisi.

OPERAATTORIEN OHJESIVUJA

Useimmilla Internet-operaattoreilla on tarjolla ohjesivu, josta löydät tarkempia tietoja kotisivun siirtämisestä WWW-palvelimelle. Alta löydät listan joistain suomalaisista operaattoreista ja heidän tarjoamistaan ohjesivuista.

Jotkut sivut ovat suojattuja, eivätkä niitä voi lukea muut kuin operaattorin varsinaiset asiakkaat. Jos käyttämääsi operaattoria ei ole mainittu listassa, voit kysyä lisätietoja operaattorisi asiakaspalvelusta.

Clinet	**http://www.clinet.fi/clinet97/clinet/www-faq.html**

Compart	**http://www.compart.fi/ohjeet/www/**
Cute Communications	**http://www.cute.fi/cute/kotisivu.html**
DLC	**http://www.dlc.fi/faq.html#ftp**
Eunet	**http://personal.eunet.fi/support/kotisivu.html**
IWN	**http://www.iwn.fi/ohjeet/koti/**
Japo	**http://www.japo.fi/ohjeet/kotisivu/aloitus.html**
Kolumbus	**http://www.kolumbus.fi/helpdesk/ohjeita/koti/index.html**
Megabaud	**http://www.megabaud.fi/ohjeet/howtopublish.html**
Netti Finland	**http://www.netti.fi/asiakaspalvelu/kotisivu.html**
PCUF	**http://www.pcuf.fi/pcuf/kotisivu.html**
Sci.fi	**http://home.sci.fi/ohjeita/**
SIP	**http://info.sip.fi/omatwww.html**
Tele iNet Pro	**http://www.inet.fi/keskuskatu/inet/kotisivu/**
WakkaNet	**http://www.wakkanet.fi/helpdesk/kotisivu.html**

LUKUOIKEUKSIEN PÄIVITTÄMINEN

Yksi mahdollisista syistä siihen, että sivusi eivät näy selaimessa saattaa olla lukuoikeuksien puuttuminen.

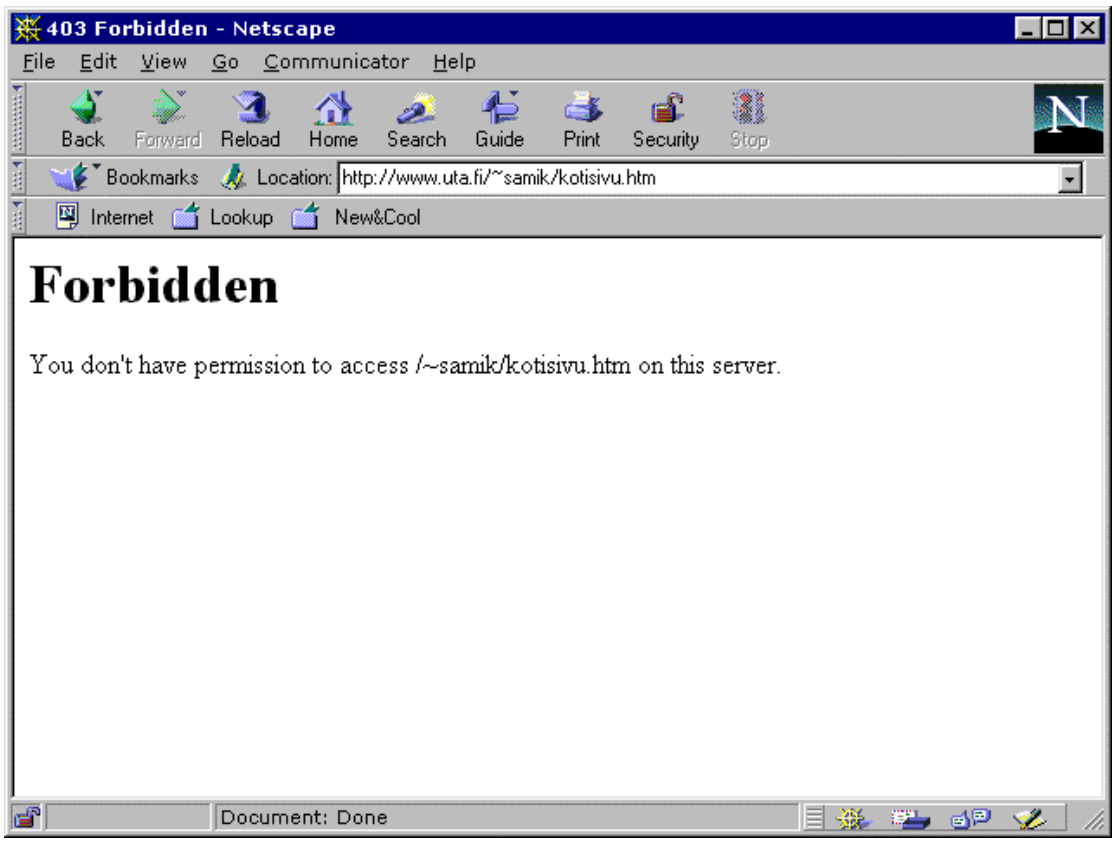
Lukuoikeuksiin törmää yleensä monen käyttäjän tietokoneissa. Näissä koneissa tiedostot suojataan siten, että käyttäjät eivät pääse lukemaan tois-

tensa tiedostoja. Yleensä tämä onkin erittäin hyvä idea (et varmaankaan halua kenenkään lukevan esimerkiksi sähköpostejasi). WWW-sivujen tapauksessa tiukoista lukuoikeuksista saattaa olla kuitenkin myös haittaa.

Kun selaimesi näyttää sinulle WWW-sivuja, se ottaa yhteyttä palvelinkoneessa ajettavaan WWW-palvelinohjelmistoon ja pyytää siltä sivua, jonka halusit nähdä. Selaimet osaavat hakea itsenäisesti tiedostoja vain oman koneesi kovalevyltä, mutta Internet-yhteyden yli sivuja luettaessa selain tarvitsee aina palvelinohjelmiston apua. Kun selain on pyytänyt palvelimelta tiettyä WWW-sivua, palvelin ottaa selville sivun sijainnin, lukee sen sisään ja välittää sen eteenpäin selainohjelmalle Internet-yhteytesi yli. Tämän jälkeen selain voi näyttää sivun aivan kuin se olisi ladattu oman koneesi kovalevyltä.

Monen käyttäjän tietokoneissa itse WWW-palvelinohjelmisto on "käyttäjä" muiden joukossa. Tämä tarkoittaa sitä, että WWW-palvelin ei pysty lukemaan tekemiäsi HTML-tiedostoja ellet anna sille erikseen lupaa. Useimpien operaattorien koneissa tämä lupa, eli lukuoikeudet, annetaan automaattisesti, niin että kotisivun tekijöiden ei tarvitse huolehtia siitä erikseen.

Joskus kuitenkin saattaa käydä niin, että lukuoikeudet puuttuvat ja WWW-palvelin ei pysty lukemaan sivujasi eikä siis antamaan niitä myöskään eteenpäin selaimellesi. Halutun sivun sijasta palvelin antaa ruudullesi virheilmoituksen:



Tämä virheilmoitus tarkoittaa siis sitä, että palvelin ei pystynyt lukemaan sivuasi lukuoikeuksien puuttumisen vuoksi.

Lukuoikeuksia on kahta tyyppiä: hakemistojen ja tiedostojen oikeuksia. Kun WWW-palvelin haluaa lukea tekemiäsi sivuja, sen täytyy ensiksi päästä tutkimaan hakemistoa, jossa tiedostot sijaitsevat. Nämä hakemistot tarvitsevat siis *selausoikeudet*. Sen lisäksi palvelimen täytyy pystyä lukemaan itse tiedostot, niin että ne voidaan antaa eteenpäin niitä pyytävälle selaimelle. Tiedostot tarvitsevat siis *lukuoikeudet*.

Lukuoikeuksien muuttamisen voit tehdä avaamalla **telnet**-yhteyden operaattorisi palvelinkoneeseen. Kun olet kirjoittautunut palvelinkoneeseen sisään, anna ensin selausoikeudet HTML-sivujasi sisältävälle hakemistolle **chmod**-komennolla:

```
chmod a+x public_html
```

Yllä olevassa esimerkissä sivujasi sisältävän hakemiston nimi on siis **public_html** (vaihda tarvittaessa tilalle oman operaattorisi käyttämä nimi). Anna sen jälkeen lukuoikeudet kaikille kyseisessä hakemistossa oleville tiedostoille:

```
chmod -R a+r public_html/*
```

Huomaa iso R-kirjain ja rivin lopussa oleva tähtimerkki. Näiden komentojen jälkeen lukuoikeuksien pitäisi olla kunnossa - jos et onnistu vieläkään saamaan sivujasi näkyviin, ota yhteyttä operaattorisi asiakaspalveluun. Operaattorillasi saattaa olla myös erillinen ohje lukuoikeuksien muuttamiseen.

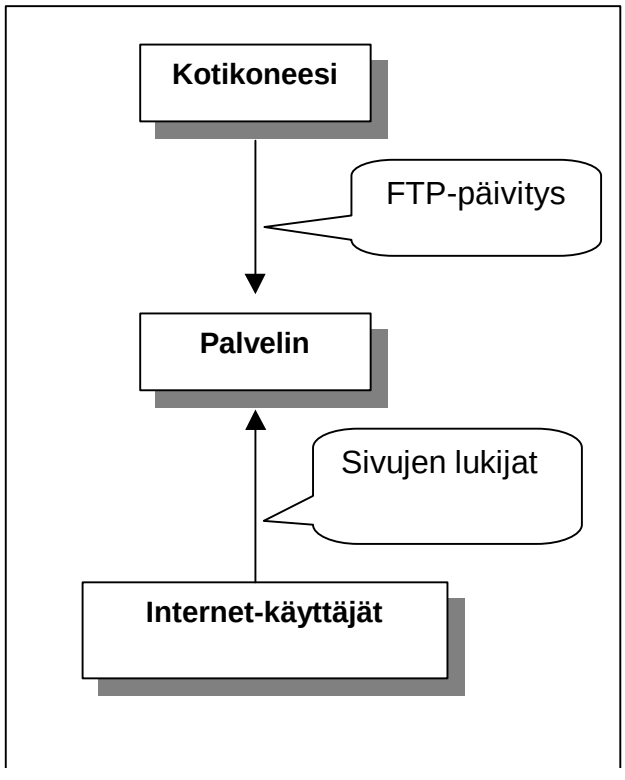
TIEDOSTOJEN KANSSA TYÖSKENTELY

Kun olet saanut sivusi onnistuneesti siirrettyä verkkoon, voit julkistaa sivujesi osoitteen tutuillesi. Käyttämäsi URL-osoite on yleismaailmallinen, eli kuka tahansa Internetin käyttäjä pystyy nyt näkemään sivusi käyttäen samaa osoitetta kuin millä sinä sait sivusi näkyviin.

Tästä eteenpäin WWW-sivuistasi on olemassa kaksi kopiota.

Ensimmäinen kopio sivuista sijaitsee oman tietokoneesi kovalevyllä. Aina kun teet sivuihisi muutoksia, muokkaat näitä sivuja. Samoin kaikkia sivuihisi liittyviä kuvia muokataan koneesi kuvalevyllä. Jos teet uusia sivuja, teet ne ensin koneesi kovalevylle, jossa myös muokkaat niitä.

Toinen kopio sivuista sijaitsee Internet-operaattorisi palvelinkoneessa. Koska sivut ovat palvelinkoneessa, niitä voidaan lukea mistä tahansa tietokoneelta, jossa on käytössä WWW-selain ja Internet-yhteys. Nämä sivut ovat ne, joita kaikki muut käyttäjät lukevat. Jokaisella palvelimessa sijaitsevalla sivulla ja kuvalla on oma URL-osoite, joten voit julkistaa tämän osoitteen muille käyttäjille. Sivuihisi voidaan tehdä myös linkkejä.



Kun olet muokannut omassa tietokoneessasi sijaitsevia WWW-sivuja ja haluat julkistaa muutokset, näistä sivuista tehdään kopio palvelinkoneeseen. Vanhemmat sivut korvaantuvat uusilla sivuilla ja sen jälkeen myös Internet-käyttäjät pystyvät näkemään uusimman version sivuistasi.

Kun myöhemmin päivität tiedostojasi, niiden siirto verkkoon tapahtuu samalla tavalla kuin äskeisissä luvuissa

kerrotulla ensimmäisellä kerralla. Tiedostojen muokkauksen vaiheet sujuvat seuraavasti:

1. Muokkaa ensin tiedostoja ja kuvia koneesi kovalevyllä. Tarkista sivusi katselemalla niitä selaimellasi tai useilla selaimilla ja tee tarvittavat muutokset. Kun sivut ovat siinä kunnossa, että olet valmis päivittämään ne verkkoon, voit siirtää ne operaattorin palvelimelle.

2. Aloita siirtäminen käynnistämällä tiedostonsiirto-ohjelma aiemmin kuvatulla tavalla. Jos käytät **WS-FTP**:tä, voit käyttää aiemmin ohjelman muistiin tallentamiasi tietoja käyttämällä asetuksille antamaasi nimeä.

3. Siirrä kaikki tiedostot koneesi kovalevyltä operaattorin palvelinkoneelle. Muista siirtää HTML-tiedostot **ASCII**-muodossa ja kuvat **Binary**-muodossa.

4. Tarkista lopuksi, että voit lukea kaikki sivusi Internet-yhteyden yli. Avaa sivusi käyttämällä oman kotisivusi URL-osoitetta ja kokeile kaikkia linkkejä. Katso että kaikki kuvat tulevat näkyviin.

SIVUNSUUNNITTELUN JATKOKURSSI

LAYOUT - SIVUN ULKOASU

Seuraavissa luvuissa esitellään komentoja, joilla voit muokata entistä monipuolisemmin sivusi ulkoasua. Opit tekemään sivuistasi värikkäitä, muokkaamaan kirjasintyylejä ja kappaleiden ulkoasuja sekä sijoittelemaan tekstiä ja kuvia haluamillesi paikoille.

Kun käytät erilaisia ruudun ulkoasuun vaikuttavia komentoja, muista suunnitella sivusi siten, että ne toimivat myös niillä käyttäjillä, jotka eivät halua tai pysty katselemaan sivuillasi olevia kuvia. Kokeile myös, että sivusi toimii vaikka käyttämäsi komennot eivät toimisikaan, sillä monilla verkon käyttäjillä on joko vanhemman mallisia tai muuten sellaisia selaimia, jotka eivät osaa toteuttaa kaikkia komentoja.

VÄRIT

Normaalisti WWW-sivun tekstin ja taustan väritys riippuu selaimesta. Useimmissa selaimissa tekstin väri on musta ja tekstin tausta joko valkoinen tai harmaa. Linkkien värinä on yleisimmin sininen.

Jos haluat vaihtaa sivulla käytettyjä värejä perusväreistä joksikin muuksi, voit lisätä sivun **<body>**-komentoon väriattribuutteja. Tällä tavalla voit antaa sivullesi taustan, tekstin ja linkkien värit.

Attribuutti **bgcolor** kertoo taustan värin. Jos haluat, että sivusi taustaväri on punainen, saat sen aikaan aloittamalla sivusi seuraavanlaisella komennolla:

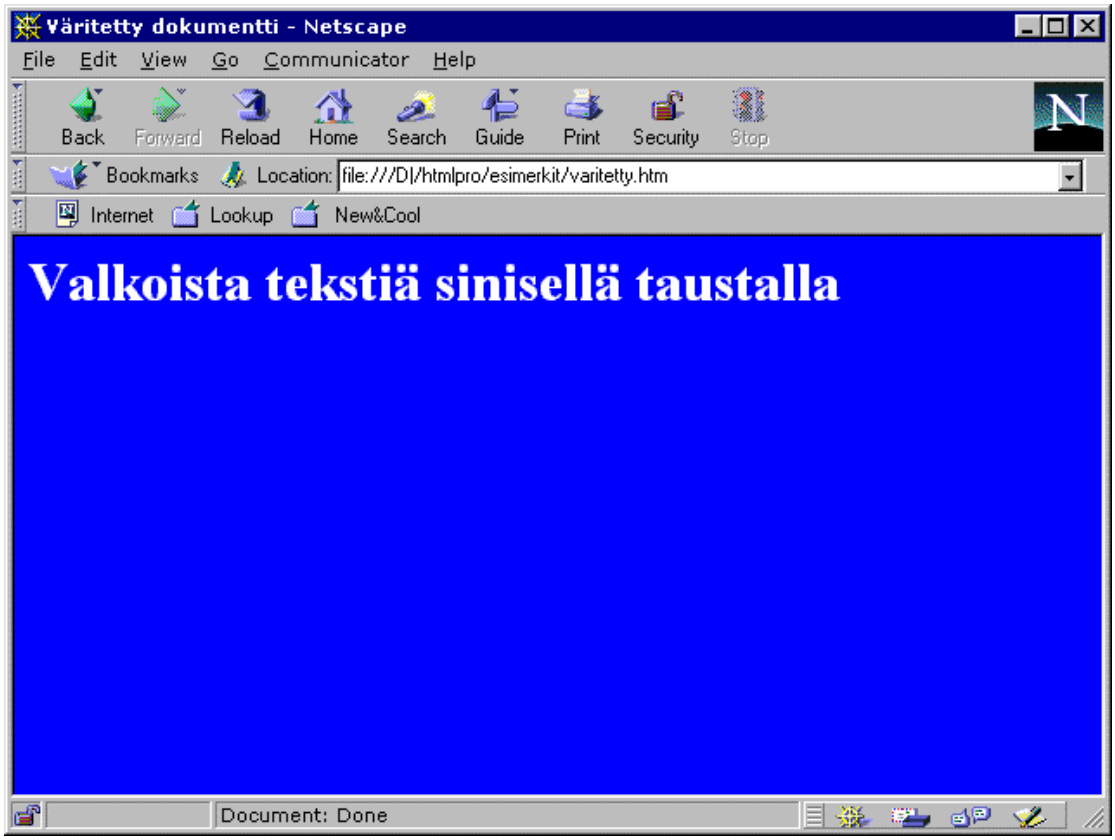
```
<body bgcolor="#FF0000">
```

HTML-värinmäärittelyt tapahtuvat antamalla värin tiedot *heksadesimaalinumerona*. Lisätietoa näistä numeroista löydät seuraavasta kappaleesta.

Attribuutilla **text** voit kertoa tekstissä käytettävän värin. Tästä on hyötyä etenkin silloin, jos olet käyttänyt taustaväriä tai -kuvaa ja sinun täytyy vaihtaa tekstin väri paremmin erottuvaksi. Saat aikaan valkoista tekstiä sinisellä taustalla aloittamalla sivun seuraavasti:

```
<html>
<head>
<title>Väritetty sivu</title>
</head>
<body text="#FFFFFF" bgcolor="#0000FF">
<h1>Valkoista tekstiä sinisellä taustalla</h1>
</body>
</html>
```

Selaimessa sivu näyttää siis seuraavanlaiselta:



Attribuutit **link**, **vlink** ja **alink** vaikuttavat linkkeihin liittyvien kuumien sanojen väritykseen. Normaalisti linkit ovat joko kirkkaan sinisiä tai violetteja, riippuen siitä oletko käynyt linkin osoittamalla sivulla vai et. Attribuutti **link** vaihtaa kuuman tekstin väriä sellaisessa linkissä, jota et ole vielä seurannut.

Niissä linkeissä, jotka olet nähnyt jo ennen, kuuman tekstin väritys on yleensä violetti. Tätä väriä voit vaihtaa attribuutilla **vlink**. Kolmas attribuutti, **alink**, kertoo aktivoidun linkin värin. Tämän värin näet silloin kun olet juuri näpäyttämässä hiirelläsi linkin tekstiä ja hiiren näppäin on alhaalla.

Jos haluat esimerkiksi vaihtaa tekstin värin punaiseksi, uusien linkkien kuumien sanojen värin vihreäksi, käytyjen linkkien värin siniseksi ja juuri valittavana olevan linkin värin violetiksi, voit käyttää seuraavanlaista komentoa sivusi alussa:

```
<body text="#FF0000" link="#00FF00" vlink="#0000FF"
alink="#FF00FF">
```

Voit antaa linkkien väriattribuutit myös englanninkielisenä tekstinä. Käytössäsi on 16 erilaista värin nimeä:

- **Black** (musta),
- **Olive** (oliivi / tumma keltainen),
- **Teal** (vihreänsininen),
- **Red** (punainen),
- **Blue** (sininen),
- **Maroon** (kastanjanruskea),
- **Navy** (laivastonsininen / tummansininen),
- **Gray** (harmaa),
- **Lime** (limetti / kirkas vihreä),
- **Fuchsia** (fuksia / kirkas violetti),
- **White** (valkoinen),
- **Green** (vihreä),
- **Purple** (purppura),
- **Silver** (hopeanharmaa),

- **Yellow** (keltainen) ja

- **Aqua** (akvamariini / kirkas sinivihreä).

Esimerkiksi, jos haluat valkoista tekstiä sinisellä taustalla, voit käyttää komentoa:

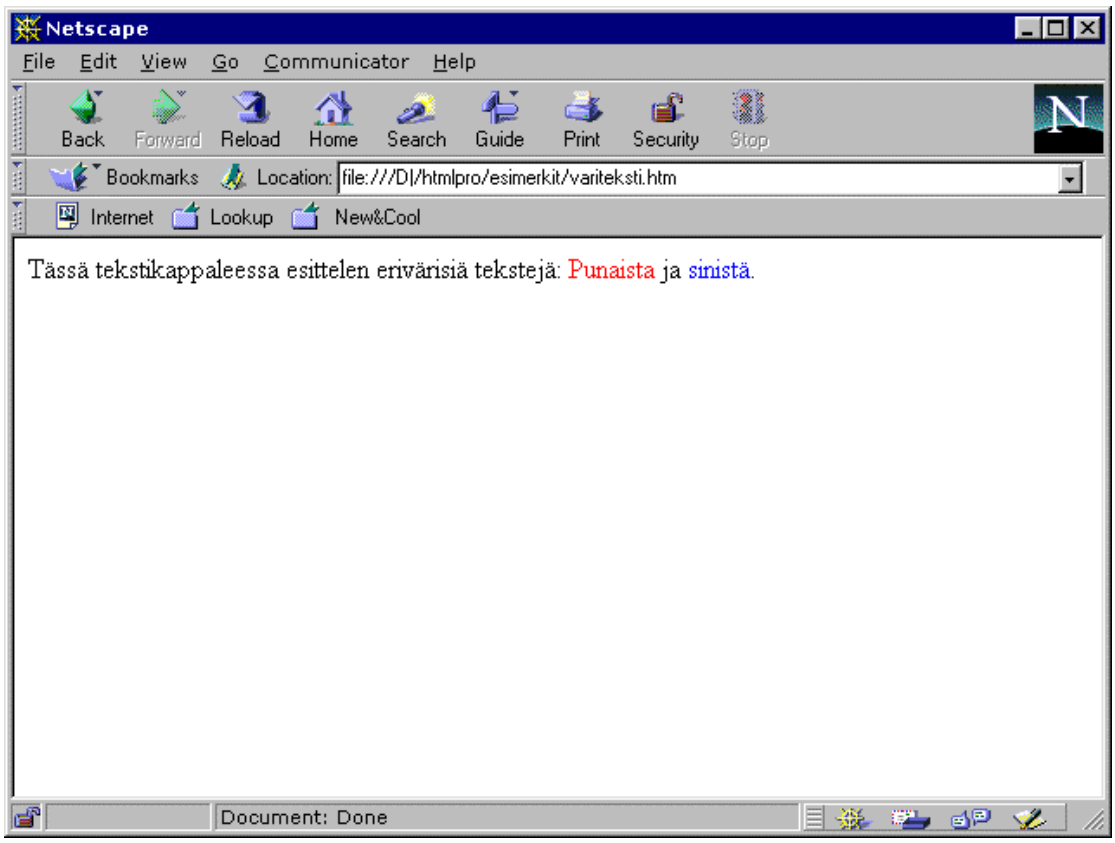
```
<body text=White bgcolor=Blue>
```

Niissä selaimissa, jotka eivät ymmärrä väriattribuutteja, tekstit näkyvät selaimen oletusarvoisilla tai käyttäjän määrittelemillä väreillä. Jos selaimesta on kytketty pois kuvien lataus, myös tekstin värit jätetään ennalleen (koska taustakuvan kanssa käytettäväksi tarkoitettu väriattribuutti saattaisi sekoittaa kuvaruudun).

Voit muuttaa myös yksittäisten tekstin kohtien värejä käyttämällä komentoa **<font>**. Kun haluat muuttaa jonkin tekstin kohdan eriväriseksi, ympäröi tämä kohta **font**-komennoilla ja lisää aloituskomentoon attribuutti **color**. Esimerkiksi näin:

```
<p>Tässä tekstikappaleessa esittelen erivärisiä
tekstejä: <font color=Red>Punaista</font> ja <font
color="#0000FF">sinistä</font>.
```

Selaimen ruudulla näet värikomentojesi tulokset:



VÄRIKOODIT

Aiemmin esitellyissä värienmäärittelyissä näkyneet erikoiset koodit ovat värien heksadesimaalikoodeja. Näissä koodeissa värien määrittelyt annetaan **RGB**-arvoina (**Red, Green, Blue**). Tämä arvo on kuuden kirjaimen mittainen merkkijono, jossa määritellään erikseen perusvärien (punaisen, vihreän ja sinisen) osuudet. Merkkijonon kaksi ensimmäistä kirjainta kertovat punaisen määrän, seuraavat kaksi vihreän määrän ja kaksi viimeistä sinisen määrän.

Värin määrä annetaan heksadesimaaliarvona: suurin arvo on **FF** ja pienin **00**. Esimerkiksi kirkkaanpunaisen RGB-arvo on **#FF0000** ja sinisen **#0000FF**. Valkoista väriä saadaan aikaiseksi sekoittamalla kaikkia värejä, eli antamalla arvo **#FFFFFF**. Musta väri syntyy silloin kun kaikki värit ovat säädetty nollaan (**#F00000**). Sekoittamalla eri sävyjä saat aikaan erilaisia värejä: esimerkiksi tummaa violettia saa aikaan sekoittamalla yhtä paljon punaista ja sinistä noin puolella täydestä arvosta (**#770077**).

Värien valitseminen sujuu parhaiten joko HTML-editorissa olevalla värinvalintatoiminnolla tai erillisellä ohjelmalla. Monet HTML-editorit sisältävät toiminnon, jolla voit määritellä värejä käyttämiisi komentoihin.

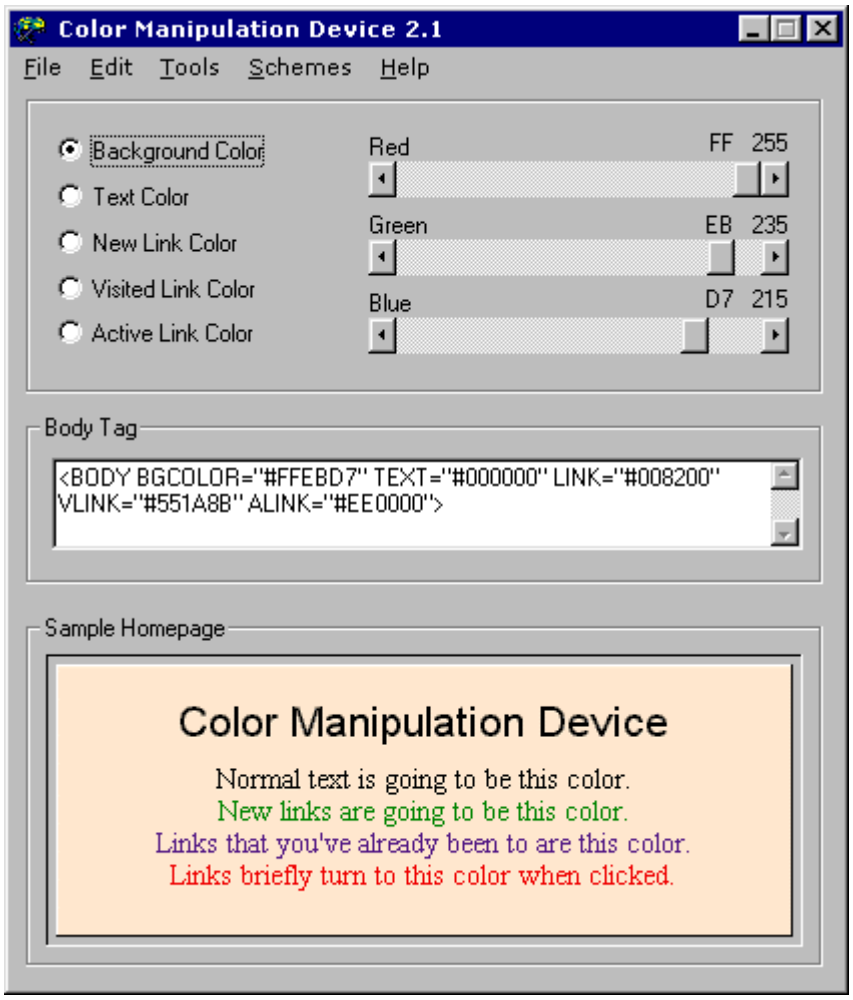
Jos käytät sivujen tekemiseen ohjelmaa, joka ei sisällä värienvalintatoimintoa, voit imuroida itsellesi Internetistä jonkin lukuisista värien suunnittelua helpottavista ohjelmista.

Yksi tällaisista ohjelmista on **Color Manipulation Device**, jonka voit kopioida itsellesi sen kotisivulta osoitteesta

```
http://www.meat.com/software/cmd/
```

Ohjelmasta on saatavilla versiot Windows 95:lle ja NT:lle sekä vanhalle Windowsille.

Kun käynnistät ohjelman, saat ruudullesi ikkunan jossa voit säätää sivun värejä:



Valitse ensin ylhäältä vasemmalta painonapeista, mitä väriä haluat säätää. Painike **Background Color** valitsee säädettäväksi sivun taustavärin. Tämän jälkeen voit siirrellä painikkeiden oikealla puolella olevia säätimiä. Säätimiä

on kolme: yksi jokaiselle päävärille (punainen, vihreä, sininen). Näitä sekoittamalla saat aikaan kaikki värit, joita tietokoneesi pystyy esittämään.

Näet jatkuvasti ruudun alareunassa esimerkin siitä, miltä värit näyttävät selaimessa. Kun olet saanut sekoitettua haluamasi värit, näet valmiin **<body>**-komennon ikkunan keskiosassa. Voit kopioida tämän komennon suoraan HTML-tiedostoosi.

Lisää värinvalintaohjelmia löydät Internetin ohjelma-arkistoista, esimerkiksi **Tucows:ista** osoitteesta

http://www.eunet.fi/tucows/color95.html

Täältä löydät myös vastaavia ohjelmia Macintoshille. Osoite on:

http://www.eunet.fi/tucows/mac/color.html

VÄRIEN KANSSA TYÖSKENTELY

Jos muokkaat sivusi värejä **body**-komennolla, on hyvä idea muuttaa aina sekä taustan että tekstin värit. Jos käyttäjä on säätänyt selaimessaan normaaliksi taustaväriksi mustan, ja muutat sivullasi tekstiväriksi mustan mutta jätät taustavärin määrittelemättä, tämä käyttäjä ei voi lukea WWW-sivuasi ollenkaan!

Kun muutat linkkien värejä, muista erottaa ne tarpeeksi hyvin tekstistä. Joillain WWW-sivuilla usein tehty virhe on muuttaa käytyjen linkkien väri samaksi kuin tavallisen tekstin väri. Tällöin käyttäjä ei enää löydä helposti sivuiltasi sellaisia linkkejä, joissa hän on käynyt. Tämä ongelma tulee esiin erityisesti silloin, jos linkit eivät näy alleviivattuina käyttäjän selaimessa.

Useimmissa selaimissa linkin oletusväri on sininen. Jos käytät sivuillasi jotain muuta väriä linkin värinä, vaarana on, että käyttäjän on paljon hankalampi huomata sivuillasi käyttämäsi linkit. Pidä linkkien väri aina sinisenä (tai sinisen sävyisenä), ellei sivujesi ulkonäkö erityisesti vaadi jotain muuta väriä.

TAUSTAKUVAT

Jos haluat elävöittää sivuasi, voit liittää siihen taustakuvan. Taustakuva tallennetaan tiedostoon ja sen täytyy olla **gif**- tai **jpeg**-muodossa oleva kuva, aivan kuten muidenkin WWW-sivulle sijoitettavien kuvien. Kun selain aloittaa sivun lataamisen, se lataa ensimmäiseksi taustakuvan ja sijoittelee sivun tekstit ja kuvat sen päälle.

Jos taustakuva on kooltaan pienempi kuin selaimen ikkuna, selain sijoittelee samaa kuvaa vierekkäin ja alekkain niin monta kertaa, että koko ikkunan tausta tulee täyteen. Kun sivua vieritetään, taustakuva siirtyy muun tekstin mukana. Alhaalta paljastuvaa tyhjää tilaa täytetään taustakuvalla.

Taustakuva määritellään lisäämällä sivun alussa olevaan **body**-komentoon attribuutti **background**. Attribuutin jälkeen sijoitetaan lainausmerkkeihin kuvan nimi, joka voi olla joko tiedoston nimi tai osoite verkossa, aivan kuten muissakin WWW-sivulle sijoitettavissa kuvissa.

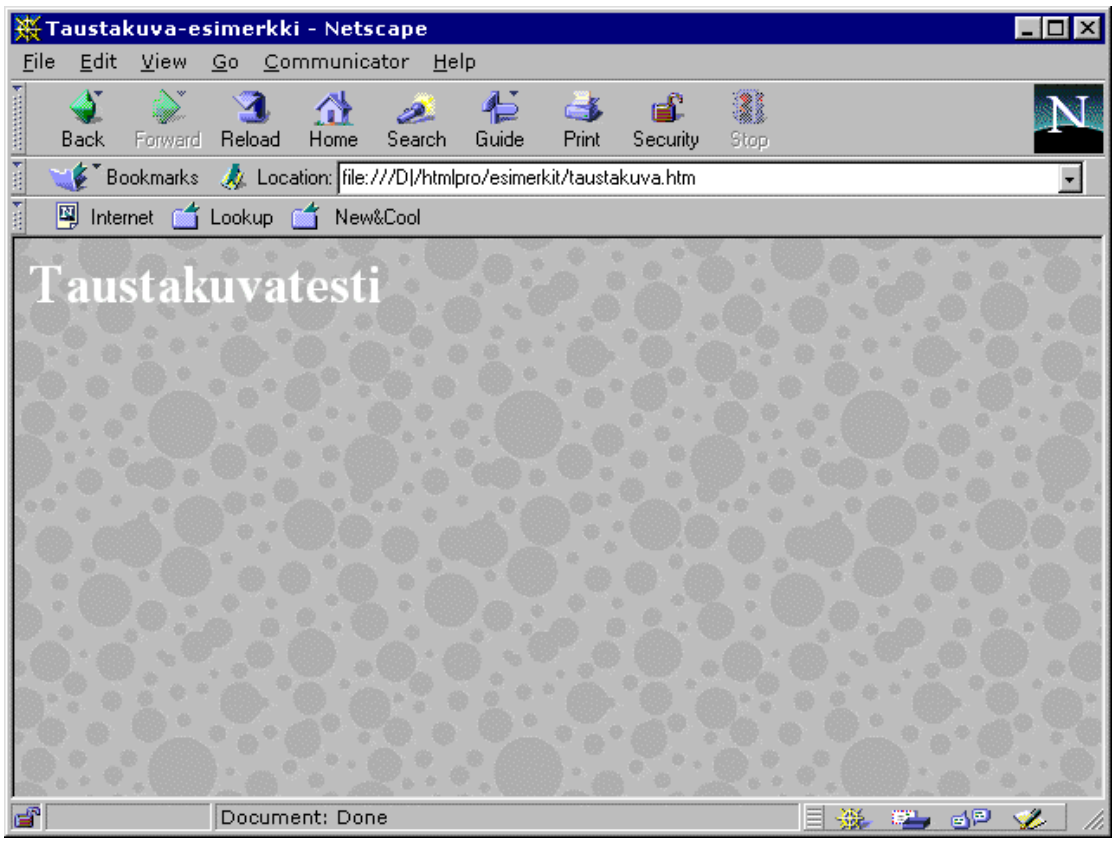
Otetaan esimerkiksi taustakuva nimeltään **grey_dots.gif**, joka löytyy Netscapen taustakuvasivulta[2]. Kuva on tallennettu HTML-tiedoston kanssa samaan hakemistoon, joten siihen voi viitata pelkällä tiedoston nimellä:

```
<html>
<head>
<title>Taustakuva-esimerkki</title>
</head>
<body background="grey_dots.gif" text=White>
<h1>Taustakuvatesti</h1>
</body>
</html>
```

Koska taustakuva on harmaasävyinen, tekstin väriksi määriteltiin valkoinen, jotta teksti erottuisi paremmin. Selaimen ruudulla näet taustakuvan käytännössä:

---

[2] **http://www.netscape.com/assist/net_sites/bg/backgrounds.html**



Koska sivulla on vain yksi **<body>**-komento, joka tulee ennen varsinaista sivun tekstiä, sivulla voi siten olla vain yksi taustakuvio.

Taustakuvaksi kannattaa valita kuva, jossa on mahdollisimman vähän kontrasteja ja vähän erottuvia yksityiskohtia. Kun taustan päälle tulostetaan tekstiä ja kuvia, räikeä taustakuva saattaa tehdä tekstin lukemisen mahdottomaksi. Yritä valita tekstisi värit sellaisiksi, että ne erottuvat mahdollisimman hyvin taustakuvan päältä.

Ota myös huomioon lukijat, joiden näyttö ei pysty näyttämään monia värejä Tarkista, että taustakuvasi ei tee tekstistä lukukelvotonta 16-värisellä tai mustavalkoisella näytöllä. Monet taustakuvat muuttuvat mustavalkoisilla näytöillä kokonaan mustiksi.

Koska taustakuvaa toistetaan pysty- ja vaakasuunnassa niin, että se täyttää koko kuvaruudun, kannattaa valita kuva, jonka reunat sopivat yhteen ja muodostavat jatkuvan kuvapinnan.

Huomaa, että taustakuvan lataaminen hidastaa aina sivusi lataamista. Selain joutuu lataamaan koko taustakuvan ennen kuin se voi aloittaa muun sivun muotoilun. Tämä hidastaa sivun latausta etenkin silloin, jos taustakuvasi tiedosto on iso. Taustakuvasta kannattaakin yrittää tehdä

mahdollisimman pieni. Kuvaan kannattaa ottaa mukaan toistuvasta kuviosta vain yhden kuvion kokoinen osa.

FONTIT

Jotkut selaimet tarjoavat mahdollisuuden muuttaa sivun tekstin kokoa. Kirjasinkokoa voi vaihtaa kesken sivun. Koska kirjasintyypit ja koot vaihtelevat käytetyn kappaletyypin mukaan (esimerkiksi otsikoissa ja leipätekstissä), kaikki kirjasinkoon muutokset tehdään suhteellisina.

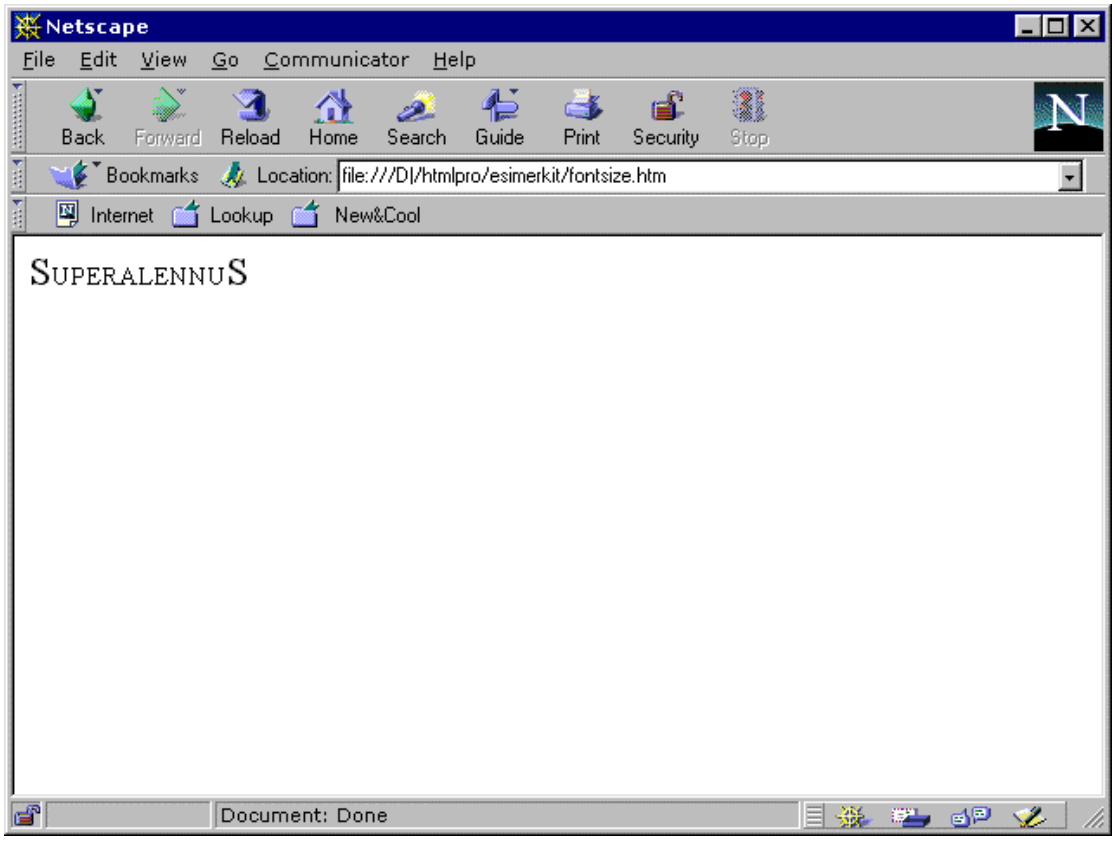
Tekstin normaali kirjasinkoko on oletusarvoisesti 3. Kirjasinkoon voi säätää välille 1-7. Tällöin kirjasinkoko 2 on hieman normaalia tekstityyppiä pienempi, kirjasinkoko 4 hieman suurempi ja 7 paljon suurempi.

Kirjasinkoko määritellään komennolla **font**. Sille voi antaa attribuutilla **size** joko uuden kirjasinkoon tai suhteellisen muutoksen edellisestä kirjasinkoosta. Suhteellinen muutos esitetään numerona, jonka edessä on joko plus-merkki (**+**) tai miinus-merkki (**-**). Lopetuskomennon jälkeen tuleva teksti on taas normaalikokoista.

Alla esimerkki tekstipätkästä, jossa ensimmäinen ja viimeinen kirjain merkitään suurempana kuin loput kirjaimet:

```
<p>
<font size=+2>S</font>
UPERALENNU
<font size=+2>S</font>
```

Selaimessa lopputulos näyttää tältä:

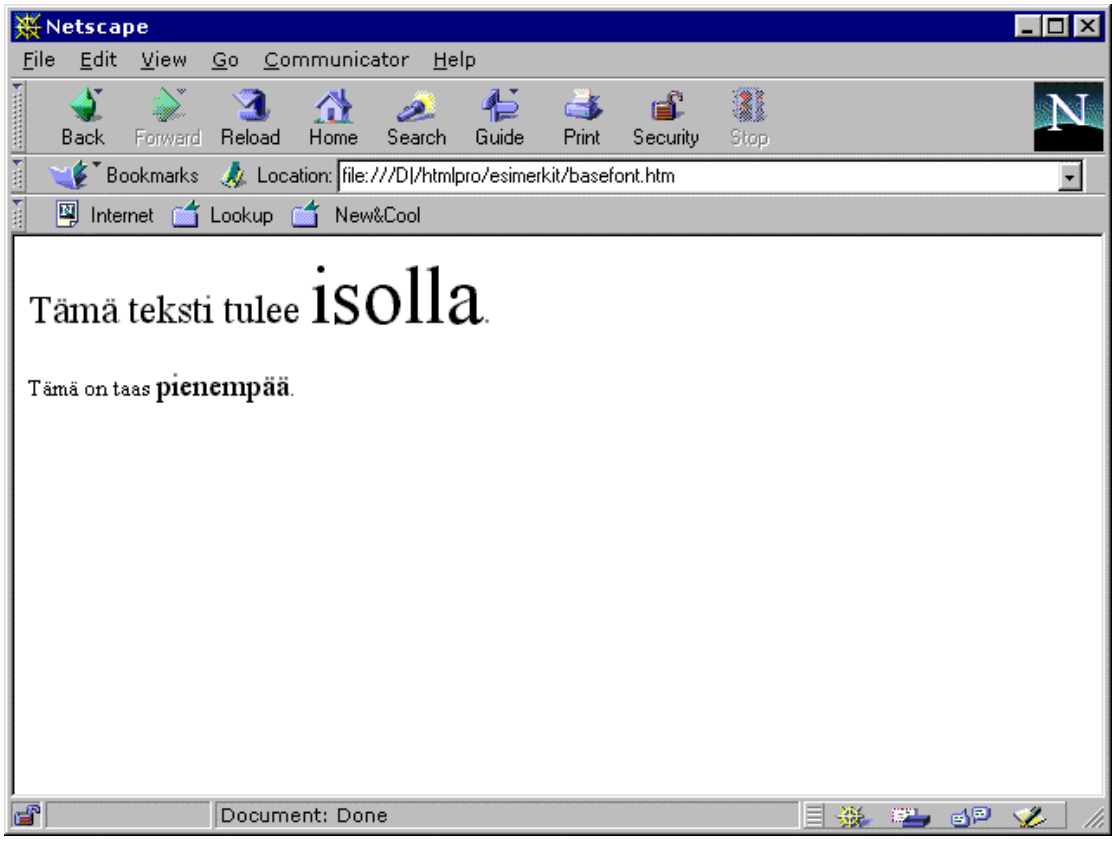


Komennolla **basefont** voit vaihtaa kirjasintyypin peruskoon numeroa, eli sitä minkä kokoisena tavallinen teksti ilman **font-**määrittelyitä tulee lukijan näytölle.

Esimerkki:

```
<basefont size=5>
<p>Tämä teksti tulee
<font size=+2>isolla</font>.
<basefont size=2>
<p>Tämä on taas
<font size=+2>pienempää</font>.
```

Kuvaruudulla:

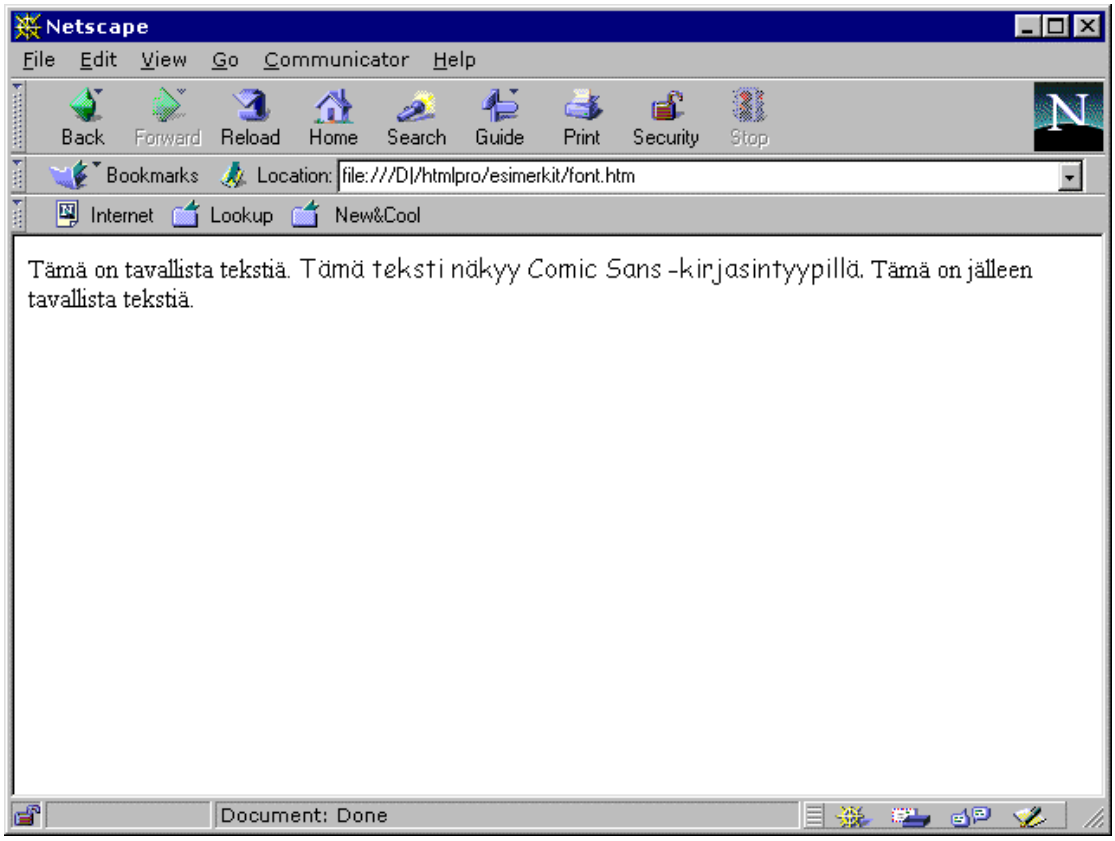


Voit määritellä myös ruudulla käytettävän fontin ulkoasun, niin ikään **font**-komennolla. Kirjasintyyppi valitaan lisäämällä **font**-komentoon attribuutti **face**.

Kirjasintyyppi on tietokoneessa sijaitsevan kirjasintyypin nimi. Jos esimerkiksi haluat, että tekstisi näkyy kuvaruudulla **Comic Sans MS**-fontilla, ympäröi tekstisi seuraavanlaisilla määrittelyillä:

```
<p>Tämä on tavallista tekstiä.
<font face="Comic Sans MS">Tämä teksti näkyy Comic
Sans -kirjasintyypillä.</font> Tämä on jälleen
tavallista tekstiä.</p>
```

Kuvaruudulla se näyttää tältä:



Jos lukijan koneessa ei ole asennettuna haluamaasi kirjasintyyppiä, voit määritellä vaihtoehtoisia kirjasintyyppejä antamalla lisää attribuutteja **<font>**-komentoon:

```
<font face="Comic Sans MS","Arial">Hip ja hop.</font>
```

Yllä oleva teksti näkyy **Comic Sans MS**-kirjasintyypillä, jos se on asennettu lukijan koneeseen. Jos tätä kirjasintyyppiä ei löydy, ohjelma yrittää näyttää tekstin **Arial**-kirjasintyypillä. Jos sitäkään ei löydy, käytetään normaalia tekstityyppiä.

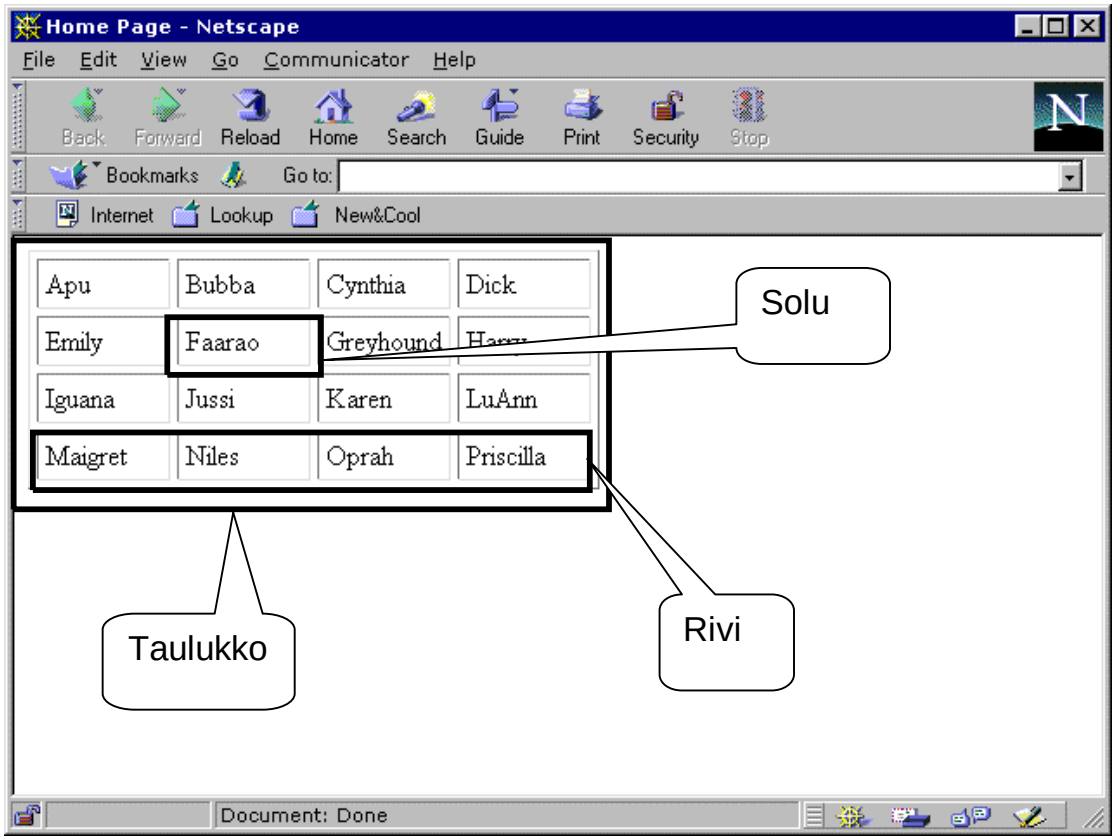
Kun suunnittelet sivujesi ulkoasua, muista aina että kaikilla lukijoillasi ei ole välttämättä käytössään haluamiasi kirjasintyyppejä. Tee sivuistasi sellaiset, että niiden toiminta ei riipu valitsemistasi kirjaintyypeistä. Kokeile että sivut näyttävät hyvältä vaikka haluamasi fontit eivät ole saatavilla, esimerkiksi muuttamalla komennoissa olevat kirjasintyyppien nimet väliaikaisesti sellaisiksi, joita koneestasi ei löydy:

```
<font face="XXComic Sans MS","XXArial">
Hip ja hop.</font>
```

TAULUKOT

Taulukkokomentojen avulla on mahdollista toteuttaa sekä erilaisia taulukoita että myös muunlaisia efektejä. Esimerkiksi palstoitus on mahdollista toteuttaa näiden komentojen avulla.

Taulukot koostuvat kolmesta osasta: itse taulukosta, sen riveistä ja soluista.



TAULUKKO

Taulukot aloitetaan **<table>**-komennolla ja lopetetaan **</table>**-komentoon. Näiden komentojen väliin sijoitetaan varsinaiset taulukon solut.

Taulukko sisältää tietyn määrän sarakkeita. Määrää ei tarvitse määritellä erikseen vaan selain laskee sen itse annettujen solujen perusteella. Taulukon sarakkeiden lukumäärä lasketaan leveimmän taulukon osan mukaan — jos taulukko sisältää rivin, jossa on kaksi solua sekä rivin, jossa on kolme solua, taulukko muotoillaan automaattisesti kolmisarakkeiseksi. Lyhyemmän rivin loppuun lisätään yksi tyhjä solu.

Taulukon leveys ja korkeus ruudulla määräytyvät solujen sisällön perusteella. Taulukosta tehdään niin leveä ja korkea kuin sarakkeiden sisällön esittämiseen tarvitaan. Jos selaimen ikkunan leveys ei riitä taulukolle, solujen sisällä olevia tekstejä jaetaan useammille riveille siten, että taulukko mahtuu kokonaan näkyviin.

Kun selaimen ikkunan kokoa muutetaan, se muokkaa taulukon uudelleen uudenkokoisen ikkunan mukaiseksi.

Sivulla voi olla useita taulukoita — jokaisen taulukon koko määräytyy erikseen sen sisällön mukaan. Taulukon solujen sisään on mahdollista sijoittaa toisia taulukoita. Tällä tavalla voit toteuttaa hieman erikoisempiakin taulukoita.

Normaalisti taulukkoon ei piirretä kehyksiä. Jos haluat nähdä reunat taulukon ympärillä sekä rivien ja sarakkeiden välissä, lisää komentoon attribuutti **`border`**.

Aletaan tekemään edellisen luvun kuvassa näkyvää taulukkoa. Taulukon tekeminen jakautuu useisiin vaiheisiin. Ensimmäiseksi lähdetään siis liikkeelle **`table`**-komennolla. Komentoon lisätään attribuutti **`border=1`**, joka kertoo, että taulukolle piirretään myös reunat:

```
<table border=1>
<!-- tähän väliin sijoitamme taulukon -->
</table>
```

Nämä komennot yksinään eivät vielä tee näkyviin taulukkoa, vaan seuraavaksi taulukkoon täytyy lisätä rivejä.

TAULUKON RIVIT

Taulukon sisällä jokainen rivi merkitään komentojen **`<tr>`** ja **`</tr>`** väliin. Rivikomentojen lukumäärä kertoo selaimelle taulukon rivien täsmällisen lukumäärän.

Esimerkkitaulukossa oli neljä riviä, joten lisätään tekemäämme esimerkkiin seuraavaksi neljä rivikomentoa:

```
<table border=1>
<tr> </tr>
<tr> </tr>
<tr> </tr>
<tr> </tr>
</table>
```

Taulukko ei näy selaimessa vieläkään, koska sen sisältämät rivit ovat tyhjiä.

TAULUKON SOLUT

Taulukon aloitus- ja lopetuskomentojen sekä rivikomentojen väliin merkitään varsinaiset solut. Solujen sisältämä teksti kirjoitetaan komentojen **<td>** ja **</td>** väliin. Solujen sisään voi sijoittaa kaikkia sivun rungossa käytettäviä HTML-komentoja — myös toisia taulukoita, kuvia tai vaikkapa lomakkeiden osia.

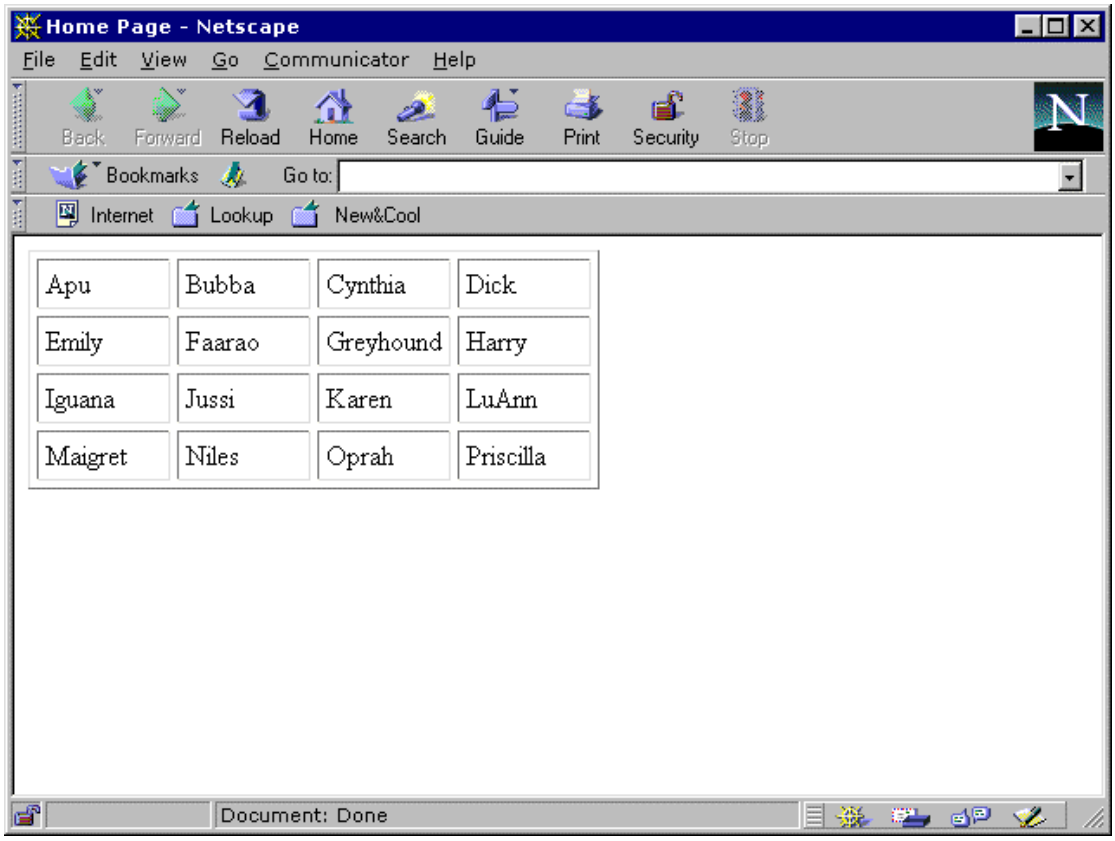
Täytetään lopuksi esimerkkitaulukkoon varsinaisten solujen sisältö. Jotta taulukkoon käytetty HTML-koodi ei muuttuisi kovin sekavaksi, jaetaan komennot eri riveille selkeiksi kokonaisuuksiksi.

```
<table border=1>

<tr>
  <td>Apu</td> <td>Bubba</td>
  <td>Cynthia</td> <td>Dick</td>
</tr>

<tr>
  <td>Emily</td> <td>Faarao</td>
  <td>Greyhound</td> <td>Harry</td>
</tr>

<tr>
  <td>Iguana</td> <td>Jussi</td>
  <td>Karen</td> <td>LuAnn</td>
</tr>

<tr>
  <td>Maigret</td> <td>Niles</td>
  <td>Oprah</td> <td>Priscilla</td>
```

```
</tr>
</table>
```

Alussa ja lopussa olevat **table**-komennot kertovat siis taulukon alun ja lopun. Taulukko sisältää neljä riviä, jotka on merkitty **tr**-komennoilla. Jokainen yksittäinen solu on merkitty **td**-komennoilla.

Selaimen ruudulla taulukko näet taulukon lopullisessa muodossaan:



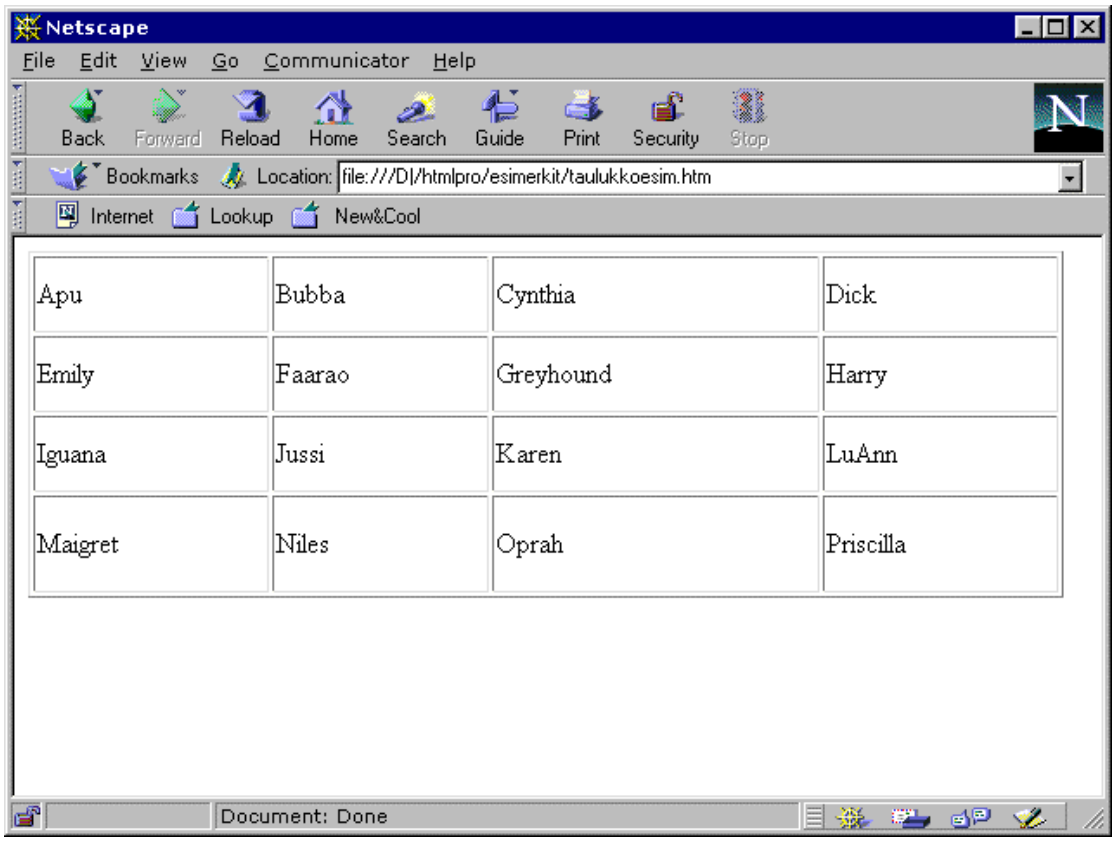
SOLUJEN KOKO

Selain tekee taulukosta senkokoisen kuin solujen esittämiseen tarvitaan. Jos haluat, että taulukko tulee ruudulle tietyn kokoisena, voit lisätä **table**-komentoon attribuutit **width** ja **height**.

Näille attribuuteille annetaan arvoksi joko prosenttiarvo selaimen ikkunan leveydestä tai koko pikseleinä. Jos haluat, että äskeisen esimerkin taulukko venyy selaimen ruudun levyiseksi ja on korkeudeltaan vaikkapa 200 pikseliä, muuta taulukon aloituskomento seuraavanlaiseksi:

```
<table border=1 width="100%" height="200">
```

Kun esimerkin muut komennot ovat samoja, taulukko tulee näkyviin selaimessa seuraavanlaisena:



Leveys- ja korkeusattribuutteja voi lisätä myös yksittäisiin soluihin, jolloin selain pyrkii muotoilemaan taulukon siten, että solu on esitetty antamiesi määrittelyiden mukaisesti.

Voit antaa taulukon tai solun kokoattribuuteissa vain toisen määrittelyistä - voit siis kertoa, että taulukon tai solun tulisi olla tietyn levyinen ja että korkeuden säätely jätetään selaimelle. Solun kokoattribuutteja ei ole pakko antaa kaikille soluille - muut taulukon solut muuttavat kokoaan tarvittaessa siten, että määrittelemäsi solu tulee näkyviin mahdollisimman paljon haluamasi kaltaisena.

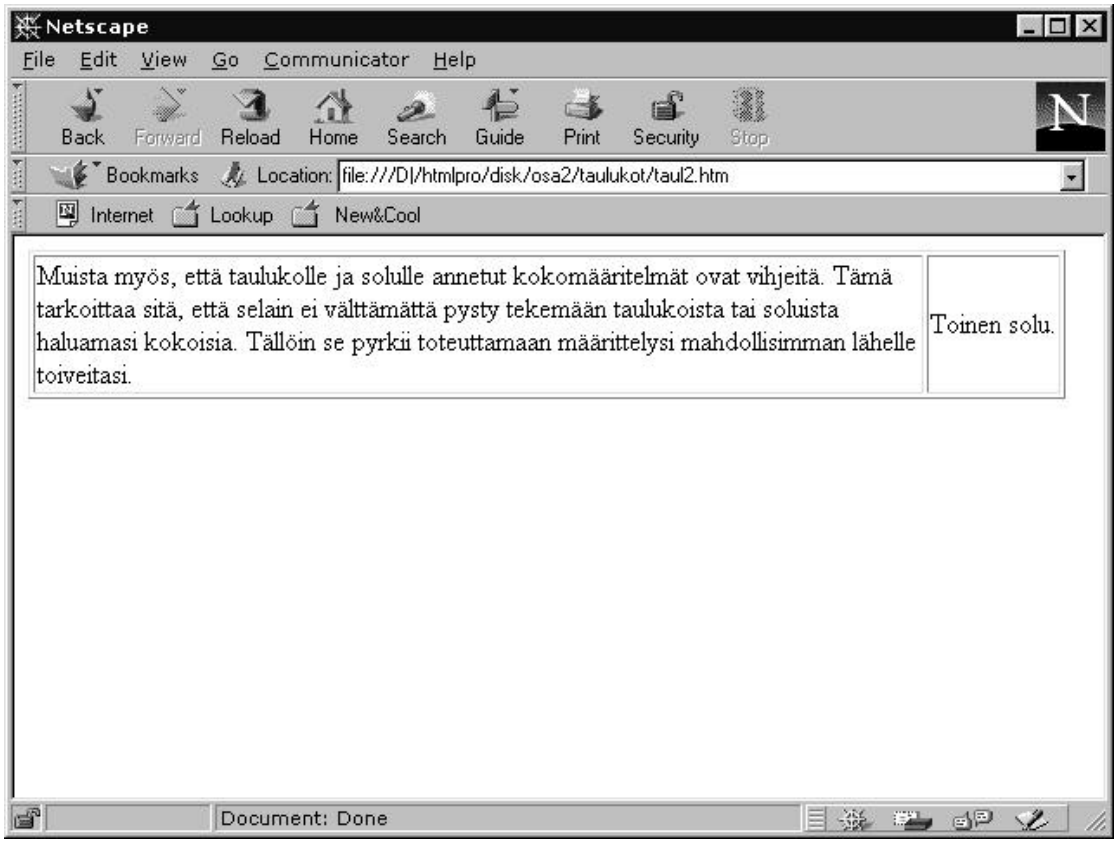
Muista kuitenkin, että jos määrittelet solun leveyden tai korkeuden, se vaikuttaa myös ympäröiviin soluihin. Kun määrittelet solulle tietyn leveyden, samassa sarakkeessa ylä- ja alapuolella olevien solujen leveys muuttuu myös, vaikka et olisikaan antanut niille attribuutteja. Jos annat ristiriitaisia attribuutteja (esimerkiksi kaksi erilevyistä solua samassa sarakkeessa), selain jättää jommankumman niistä huomiotta.

Muista myös, että taulukolle ja solulle annetut kokoattribuutit ovat vihjeitä. Tämä tarkoittaa sitä, että selain ei välttämättä pysty tekemään taulukoista tai soluista haluamasi kokoisia. Tällöin se pyrkii toteuttamaan koon mahdollisimman lähelle toiveitasi.

Jos solujen koon määrittely on jätetty selaimen huoleksi, se määrittelee solujen koon suhteessa siihen, miten paljon soluissa on tekstiä ja muuta sisältöä. Alla näet esimerkin yksirivisestä taulukosta, jonka toisessa solussa on paljon tekstiä ja toisessa vain muutama sana:

```
<table border=1>
<tr>
<td> Muista myös, että taulukolle ja solulle annetut
kokomääritelmät ovat vihjeitä. Tämä tarkoittaa sitä,
että selain ei välttämättä pysty tekemään taulukoista
tai soluista haluamasi kokoisia. Tällöin se pyrkii
toteuttamaan määrittelysi mahdollisimman lähelle
toiveitasi.</td>
<td>Toinen solu.</td>
</tr>
</table>
```

Selaimessa näet, miten tekstimäärä vaikuttaa solun kokoon:



Antamalla toiselle solulle leveysattribuutiksi 50% selaimen ikkunan leveydestä voit määritellä molemmat sarakkeet samankokoisiksi (kun annat kokoattribuutin toiselle solulle, jäljelle jäävä solu saa loput leveydestä, eli toiset 50%):

```
<table border=1>
<tr>
<td width="50%"> Muista myös, että taulukolle ja
solulle annetut kokomääritelmät ovat vihjeitä. Tämä
tarkoittaa sitä, että selain ei välttämättä pysty
tekemään taulukoista tai soluista haluamasi kokoisia.
Tällöin se pyrkii toteuttamaan määrittelysi
mahdollisimman lähelle toiveitasi.</td>
<td>Toinen solu.</td>
</tr>
</table>
```

Selaimessa näet, miten leveysattribuutti vaikuttaa taulukon ulkoasuun:

Voit myös hienosäätää solujen välistä tilaa. Solujen välissä olevaa tilaa säädetään attribuutilla **`cellspacing`**. Attribuutti **`cellpadding`** kertoo, miten paljon tilaa jätetään solun seinämien ja tekstin väliin. Nämä määrittelyt sijoitetaan taulukon aloituskomentoon. Esimerkiksi komento **`<table cellspacing=5 cellpadding=3>`** kertoo, että taulukon solujen välissä on viisi pikseliä tyhjää ja solun seinien ja tekstin välissä on kolme pikseliä tyhjää tilaa.

Attribuutilla **`border`** voi määrätä taulukon reunuksien paksuuden. Tällöin ohjelma suurentaa taulukon reunusta annetun numeron verran. Esimerkiksi **`<table border=5>`** tekee taulukon, jolla on viiden pikselin paksuinen reunus.

SOLUJEN SISÄLLÖN SIJOITTELU

Katsoessasi aiempia taulukkoesimerkkejä huomaat, että solujen sisällä oleva teksti on sijoitettu aina solun vasempaan reunaan solun keskivaiheille. Voit muuttaa halutessasi tätä sijoittelua **`valign`**- ja **`align`**-attribuuteilla. Näitä attribuutteja voi sijoittaa joko **`tr`**-komentoon (jolloin ne vaikuttavat kaikkiin sillä rivillä oleviin soluihin) tai **`td`**-komentoon (jolloin vaikutus ulottuu kyseiseen soluun).

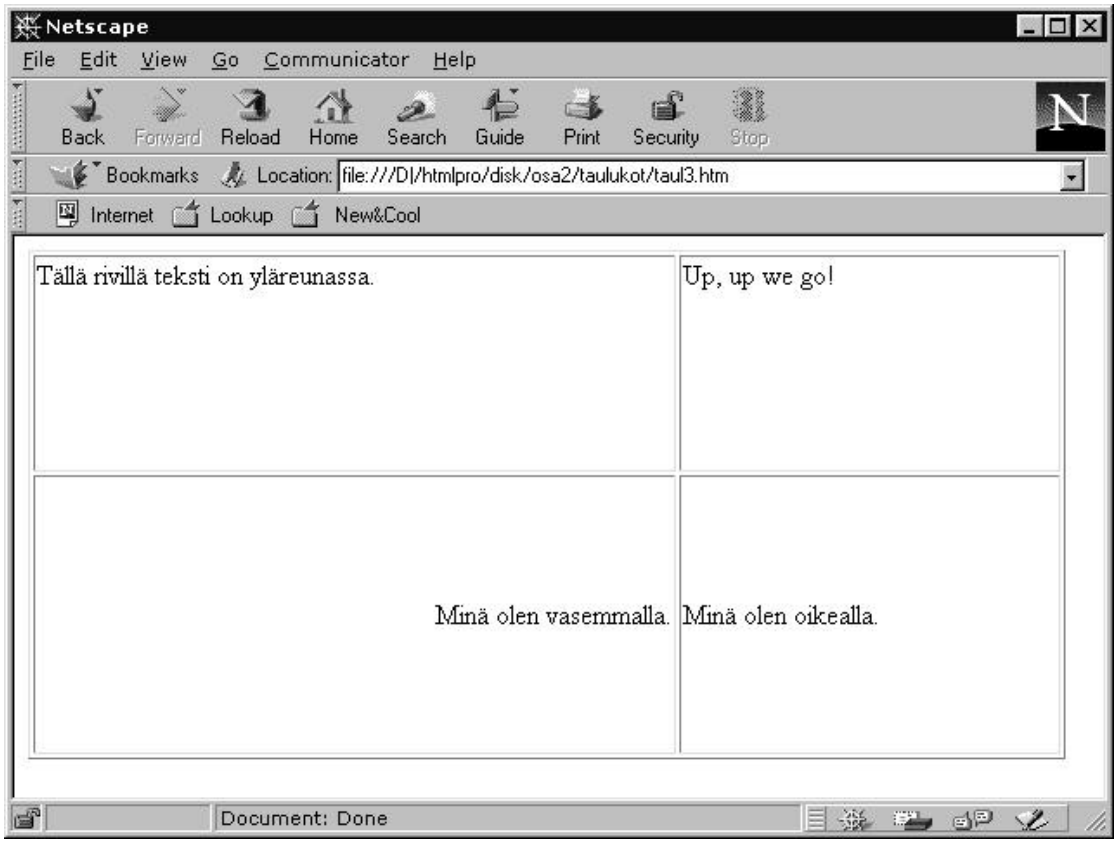
Attribuutti **valign** kertoo, miten solun sisällä oleva teksti asetellaan pystysuunnassa. Vaihtoehtoja ovat **top** (yläreunaan), **bottom** (alareunaan) ja **middle** (keskelle). Voit esimerkiksi määritellä, että rivillä sijaitsevien solujen tekstin täytyy sijaita solun yläreunassa komennolla **<tr valign=top>**.

Solujen tekstin sivusuuntainen asettelu määrätään **align**-attribuutilla. Vaihtoehtoja ovat **left** (teksti tasataan vasempaan reunaan), **right** (teksti tasataan oikeaan reunaan) ja **center** (teksti keskitetään solun keskelle).

Alla näet esimerkin solujen sisällön sijainnin määrittelyistä. Taulukko on määritelty koko ruudun kokoiseksi, jotta sijaintimuutokset näkyisivät selkeästi.

```
<table border=1 width="100%" height="100%">
<tr valign=top>
<td>Tällä rivillä teksti on yläreunassa.</td>
<td>Up, up we go!</td>
</tr>
<tr>
<td align=right>Minä olen vasemmalla.</td>
<td align=left>Minä olen oikealla.</td>
</tr>
</table>
```

Selaimessa näet lopputuloksen:



Jos rivillä olevissa varsinaisissa soluissa on annettu **valign**- tai **align**-attribuutti, ne kumoavat rivikomennossa annetun attribuutin. Esimerkiksi jos rivin alussa on määrätty solut tasattavaksi vasempaan laitaan mutta itse solussa on määritelty teksti sijoitettavaksi keskelle, solussa annettu määrittely voittaa ja teksti sijaitsee solussa keskellä.

Jos sijoitusattribuutteja ei anneta lainkaan, selain käyttää oletusarvoisia attribuutteja **valign=middle** ja **align=left** (teksti sijaitsee solun keskellä tasattuna vasempaan reunaan).

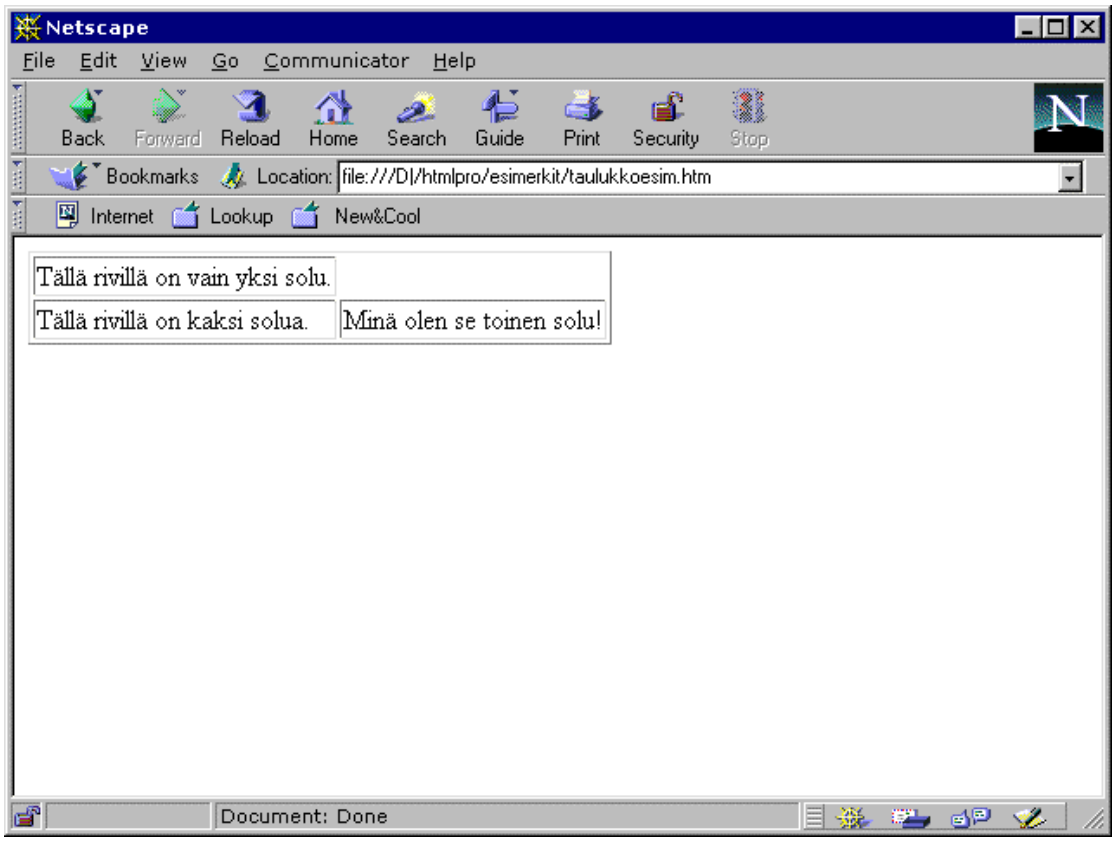
SOLUJEN SISÄLLÖSTÄ

Selain voi tarvittaessa katkaista solussa olevan tekstin välilyöntien kohdalta saadakseen taulukon mahtumaan paremmin selaimen näytölle. Jos solussa oleva teksti täytyy saada mahtumaan yhdelle riville, käytä **<td>**-komennossa attribuuttia **nowrap**. Käytä sitä kuitenkin säästeliäästi, sillä taulukosta saattaa tulla erittäin leveä, jos solut sisältävät pitkiä sanoja.

Jos et määrittele kaikkia soluja **<td>**-komennolla, selain tunnistaa määrittelemättömät solut ja sijoittaa niiden paikalle tyhjän solun:

```
<table border=1>
<tr>
<td>Tällä rivillä on vain yksi solu.</td>
</tr>
<tr>
<td>Tällä rivillä on kaksi solua.</td>
<td>Minä olen se toinen solu!</td>
</tr>
</table>
```

Sama selaimessa:



SUURET SOLUT

Kun teet taulukoita, törmäät varmasti pian tilanteeseen, jossa haluat tehdä useamman rivin tai sarakkeen kokoisia soluja. Näitä saat aikaan yhdistelemällä soluja.

Jos haluat yhdistää soluja vaaka- tai pystysuunnassa, voit käyttää **rowspan**- ja **colspan**-attribuutteja. Määrittelyllä **colspan** voit venyttää solua vaakasuunnassa — esimerkiksi attribuutti **colspan=2** kertoo, että solu

vie vaakasuunnassa kaksi saraketta. Attribuutti **rowspan** toimii samalla tavalla mutta pystysuunnassa.

Oletetaan, haluat tehdä WWW-sanomalehden. Haluat sijoittaa sivun yläreunaan otsikon ja tehdä sen alle uutisia kahdelle palstalle. Sivun tulisi näyttää tämäntapaiselta:

Kun tutkit taulukkoa, huomaat että se koostuu kahdesta rivistä ja kahdesta sarakkeesta. Poikkeuksena perinteiseen taulukkoon on, että ylimmäisen rivin solu on määritelty kahden sarakkeen kokoiseksi.

Alla näet HTML-koodin, jolla taulukko on toteutettu:

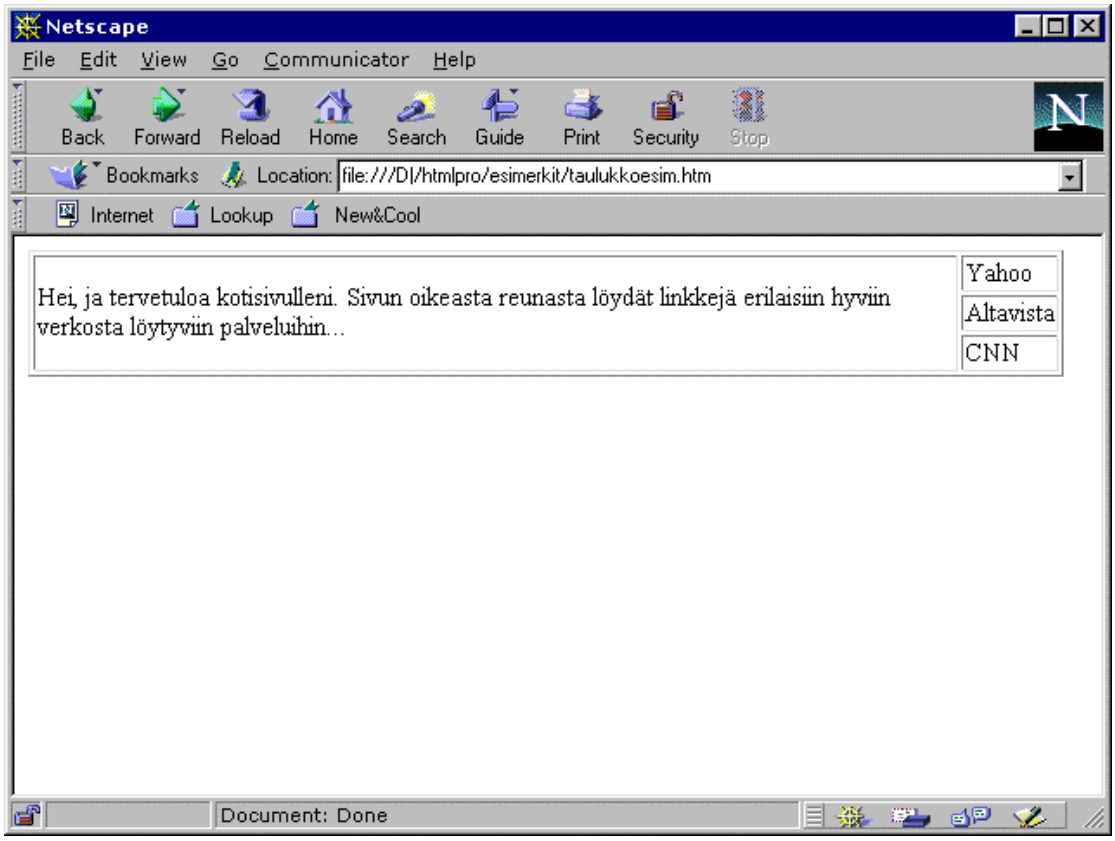
```
<table border=1>
<tr>
<td colspan=2 align=middle valign=center>
<h1>Sanomalehti</h1>
<p>Tiistaina, 21.10.1997
</td>
<tr>
<td valign=top align=left>
<h2>UFO laskeutui eduskuntatalon eteen</h2>
```

```
<p>Sikarinmuotoinen avaruusalus laskeutui tänään
eduskuntatalon eteen. Helsingin poliisi piiritti
aluksen välittömästi. Kansanedustajat eivät halua
vielä kommentoida tilannetta.
</td>
<td valign=top align=left>
<h2>Microsoft julkisti Windows 2000 -
käyttöjärjestelmän</h2>
<p>Mikrotietokoneiden uusin käyttöjärjestelmä, Windows
2000, julkistetaan ensi vuonna. Microsoft ei vahvista
huhuja, joiden mukaan käyttöjärjestelmä tarjoaisi
täydellisen moniajon jo 512 gigatavun
keskusmuistilla...
</td>
</tr>
</table>
```

Samalla tavalla voit tehdä soluja, jotka ovat kooltaan usean rivin kokoisia käyttäen attribuuttia **rowspan**:

```
<table border=1>
<tr>
<td rowspan=3>
<p>Hei, ja tervetuloa kotisivulleni. Sivun oikeasta
reunasta löydät linkkejä erilaisiin hyviin verkosta
löytyviin palveluihin...
</td>
<td>Yahoo</td>
<tr>
<!--huom: tällä rivillä ei ekaa solua -->
<td>Altavista</td>
</tr>
<tr>
<!--huom: tällä rivillä ei ekaa solua -->
<td>CNN</td>
</tr>
</table>
```

Selkeyden vuoksi esimerkistä on jätetty linkkikomennot pois. Selaimen ruudulla esimerkki näyttää tältä:



Huomaa, että toisella ja kolmannella rivillä ei määritellä ollenkaan rivin ensimmäistä solua, koska ensimmäisellä rivillä isoksi määritelty solu vie ensimmäisen solun paikan.

Selain laskee taulukon rivien lukumäärän **<tr>**-komennoista. Jos yrität määritellä **rowspan**-attribuutilla soluja, jotka menevät taulukon pohjan "läpi", ne eivät tule näkyviin.

Älä myöskään määrittele samaa solua kuuluvaksi kahteen soluun. Se voi tapahtua määrittelemällä esimerkiksi **rowspan**ia ja **colspan**ia käyttäen solun liitetyksi sekä yläpuolella että vasemmalla olevaan soluun. Riippuu käytetystä selaimesta, miltä tämä näyttää kuvaruudulla. Jotkut selaimet näyttävät näin määritellyt solut, jotkut saattavat mennä sekaisin tästä määrittelystä.

TAULUKON OTSIKKOSOLUT

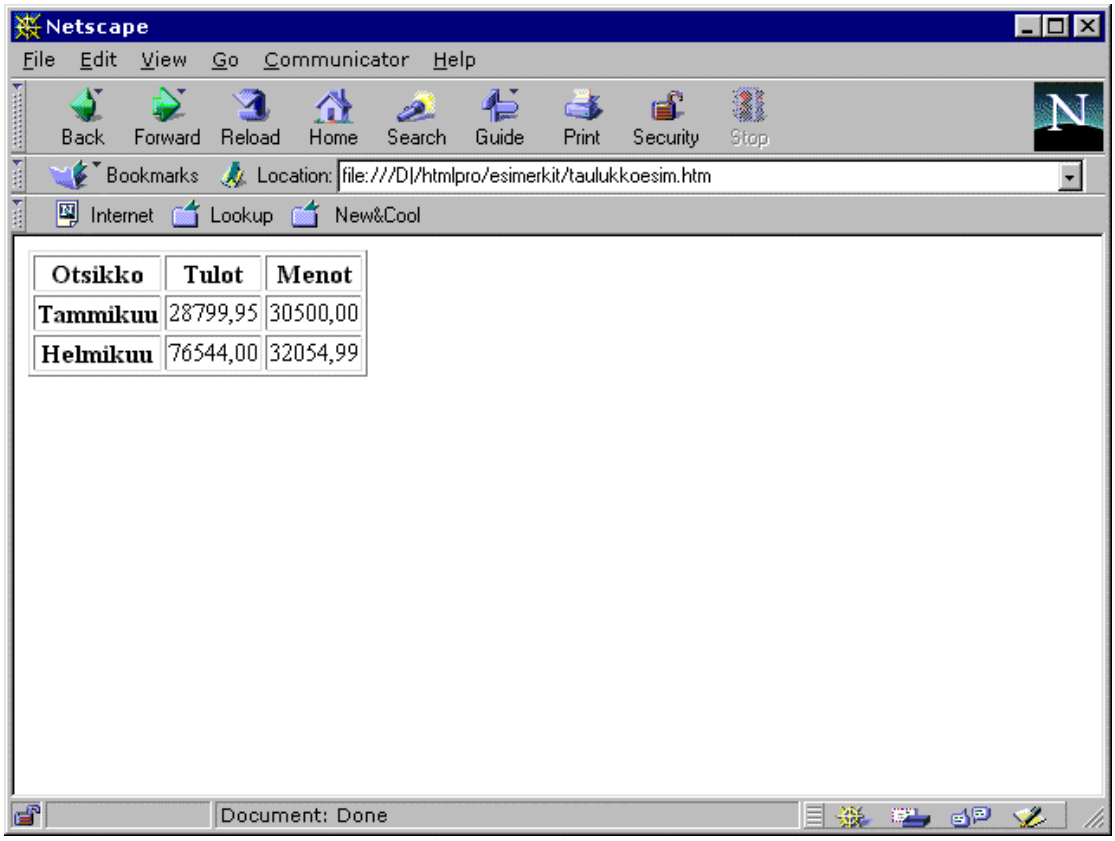
Voit tehdä otsikkosoluja sekä sarakkeisiin että riveihin. Otsikkosolut sijoitetaan taulukkoon samalla tavalla kuin tavalliset solutkin, mutta niiden merkitsemiseen käytetään **<th>**-komentoa. Komennon attribuutit ovat samanlaisia kuin **<td>**-komennossa - ainoastaan kuvaruudulla näkyvä solu-

jen asettelu ja tekstityyppi ovat erilaisia. Useimmissa selaimissa otsikkosolujen teksti on keskitetty (**align=center**) ja lihavoitu.

Rakennetaan taulukko, joka sisältää kotibudjetin. Se sisältää kolme saraketta: otsikkosarakkeen, tulot ja menot. Pystysuunnassa taulukko koostuu otsikkorivistä, tammikuun tuloista ja menoista sekä helmikuun tuloista ja menoista.

```
<table border=1>
<tr>
<th>Otsikko</th>
<th>Tulot</th>
<th>Menot</th>
</tr>
<tr>
<th>Tammikuu</th>
<td>28799,95</td>
<td>30500,00</td>
</tr>
<tr>
<th>Helmikuu</th>
<td>76544,00</td>
<td>32054,99</td>
</tr>
</table>
```

Kuvaruudulla taulukko näyttää tällaiselta:



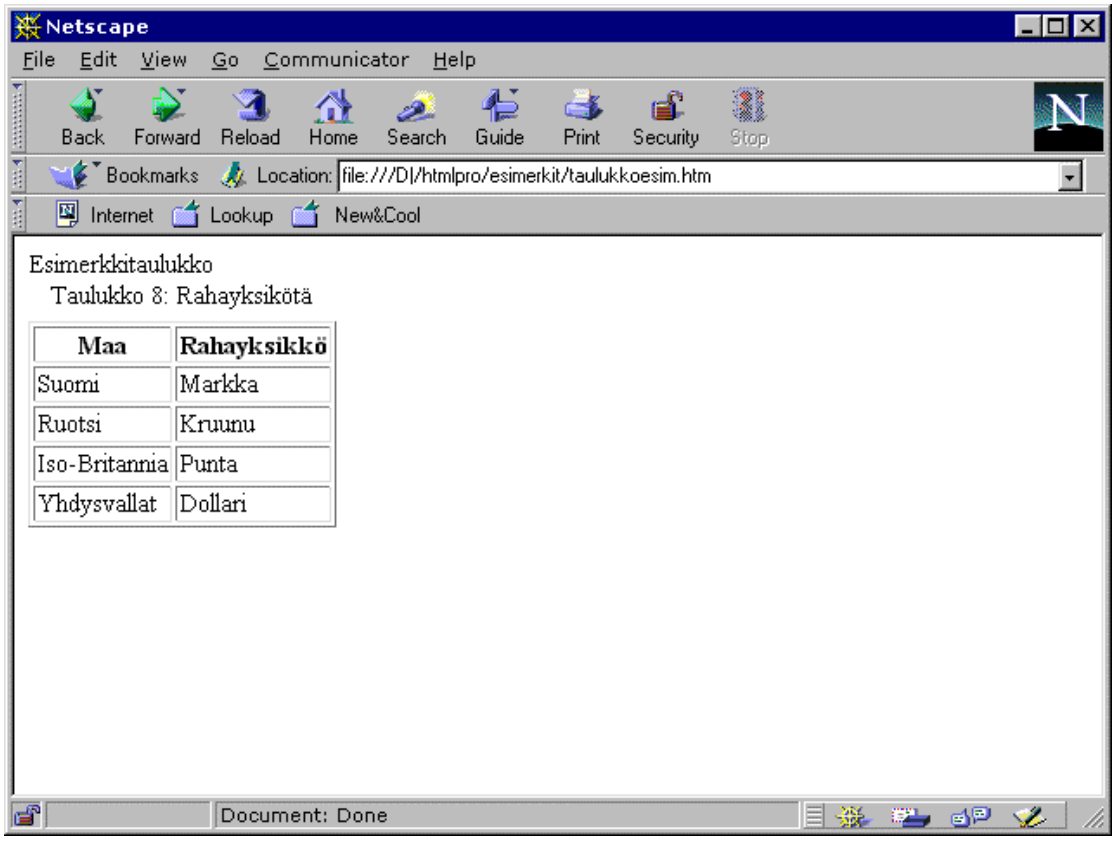
TAULUKON OTSIKKO

Jos haluat taulukolle otsikkotekstin, voit antaa sen taulukon sisällä **<caption>**-komennolla. Komento ei ole välttämätön. Se sijaitsee koodissa heti **<table>**-komennon jälkeen. **Caption**-komento tuo oletusarvoisesti taulukon yläpuolelle kuvaruudulle keskitetyn otsikon. Otsikon paikkaa voi vaihtaa taulukon alle lisäämällä komentoon attribuutti **align=bottom**.

Esimerkki taulukon otsikon määrittelystä:

```
<p>Esimerkkitaulukko
<table border=1>
<caption>Taulukko 8: Rahayksiköitä</caption>
<tr><th>Maa</th><th>Rahayksikkö</th></tr>
<tr><td>Suomi</td><td>Markka</td></tr>
<tr><td>Ruotsi</td><td>Kruunu</td></tr>
<tr><td>Iso-Britannia</td><td>Punta</td></tr>
<tr><td>Yhdysvallat</td><td>Dollari</td></tr>
</table>
```

Sama selaimessa:



SOLUJEN VÄRITYS

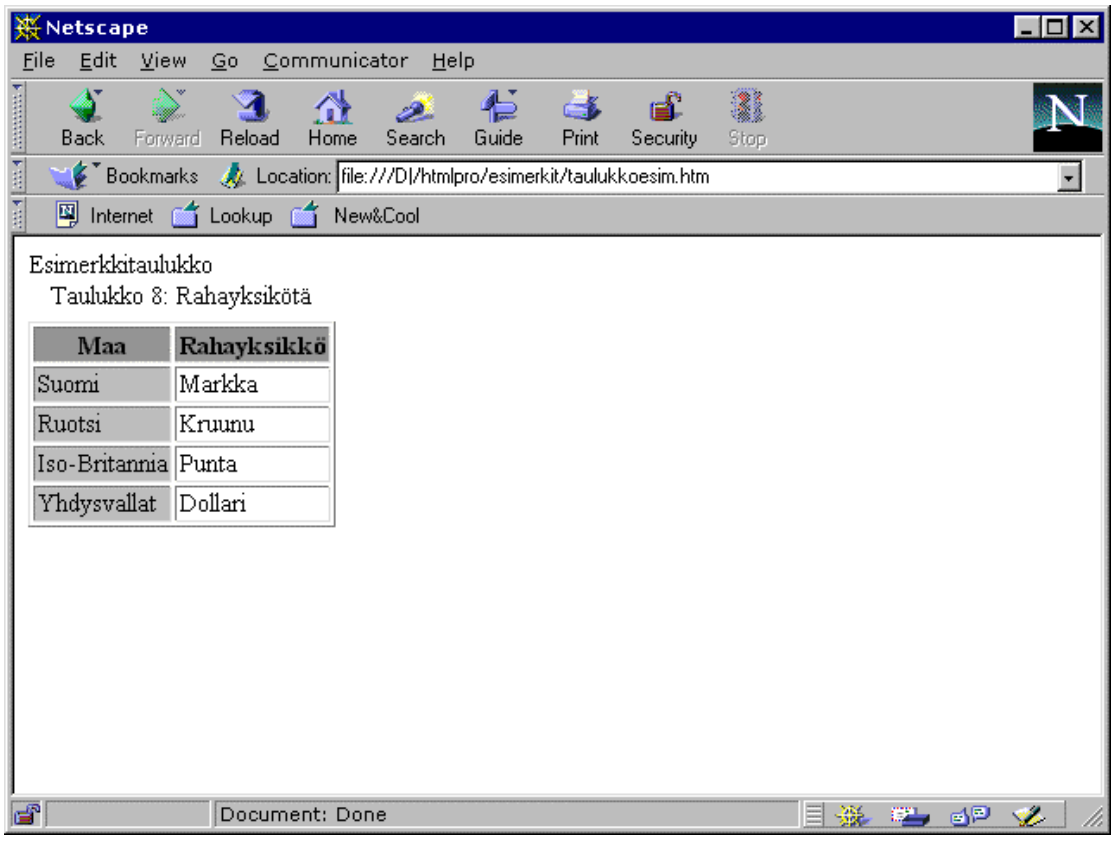
Soluille on mahdollista tehdä taustavärejä ja -kuvia, aivan kuten varsinaisille HTML-sivuillekin. Solun taustaväri määritellään attribuutilla **bgcolor**. Tämän attribuutin voi sijoittaa **table**-komentoon (vaihtaa kaikkien taulukon solujen värin), **tr**-komentoon (vaihtaa kyseisen rivin solujen värin) tai **td**/**th**-komentoon (vaihtaa kyseisen solun värin).

Solujen värit voi antaa joko värikoodeina tai värien niminä (kuten sivun värejä määritellessä). Alla näet väritetyn esimerkin rahayksikkötaulukosta:

```
<p>Esimerkkitaulukko
<table border=1>
<caption>Taulukko 8: Rahayksiköitä</caption>
<tr bgcolor="#999999">
<th>Maa</th><th>Rahayksikkö</th></tr>
<tr><td bgcolor="Silver">Suomi</td>
<td>Markka</td></tr>
<tr><td bgcolor="Silver">Ruotsi</td>
<td>Kruunu</td></tr>
<tr><td bgcolor="Silver">Iso-Britannia</td>
```

```
<td>Punta</td></tr>
<tr><td bgcolor="Silver">Yhdysvallat</td>
<td>Dollari</td></tr>
</table>
```

Väritetty esimerkki näyttää selaimessasi tältä:

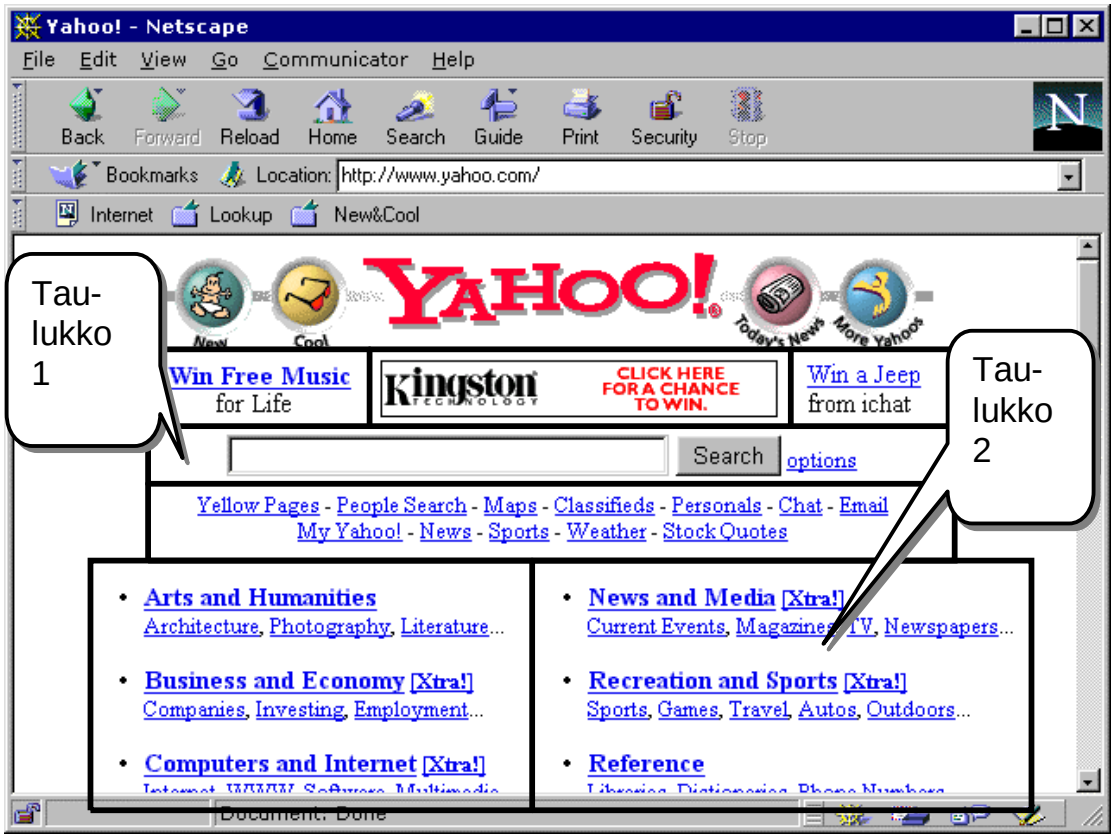


Taulukon soluihin voi sijoittaa myös taustakuvan, samalla tavalla kuin varsinaiselle sivullekin. Taustakuva tehdään sijoittamalla attribuutti **background** joko **table**-, **tr**-, **th**- tai **td**-komentoon. Attribuutin jälkeen sijoitetaan joko kuvatiedoston nimi tai sen URL-osoite.

TAULUKOT KÄYTÄNNÖSSÄ

Taulukoiden avulla saat muokattua sivujesi ulkoasua hyvin monipuolisesti. Moni Internetistä löytyvä WWW-palvelu käyttääkin taulukoita sivujensa ulkoasun määrittelyyn.

Alla näet esimerkin **Yahoo**n kotisivusta (**http://www.yahoo.com**), jossa on käytetty taulukoita sivun ulkoasun muokkaukseen. Kuvan päällä olevat katkoviivat näyttävät sinulle käytettyjen taulukoiden toiminnan:



Yahoon etusivu koostuu sivun yläreunassa olevasta otsikkokuvasta ja taulukoista. Ensimmäinen taulukko (kuvan alla) sisältää kolme riviä, joista ensimmäisessä on kolme solua. Solujen sisältö on sijoitettu **`align`**-attribuuteilla solun keskelle. Kaksi alemman rivin solua on määritelty **`colspan`**-attribuuteilla kolmen sarakkeen levyisiksi.

Alempi taulukko sisältää kaksi suurta solua, joihin **Yahoon** sisällysluettelo on sijoitettu. Itse sisällysluettelon pääkohdat on toteutettu ranskalaisilla viivoilla, joista voit lukea lisää seuraavasta luvusta.

Kun teet taulukoita sivuillesi, testaa ne ensin kytkemällä taulukon reunat päälle **`border`**-attribuutilla. Usein lopullisessa tuloksessa ei kuitenkaan tarvita taulukon reunoja, joten voit lopuksi julkaistessasi sivuja ottaa **`border`**-attribuutin kokonaan pois tai vaihtaa sen muotoon **`border=0`**.

Kun jaat tekstiä palstoihin, pidä mielessäsi kuitenkin käyttömukavuus: jos sivu on jaettu kapeisiin mutta pitkiin palstoihin, joita lukiessa sivua täytyy vierittää ylös ja alas, sen käyttömukavuus ei ole erityisen hyvä.

Jos taulukkosi eivät tunnu toimivan, tarkista seuraavat asiat:

- Onko taulukollasi aloitus- ja lopetuskomento? Jos jompikumpi näistä puuttuu, taulukkosi saattaa jäädä kokonaan näkymättä tai sen osat sijoittuvat ympäri ruutua.

- Oletko käyttänyt lopetuskomentoja säännönmukaisesti? Periaatteessa rivien ja solujen lopetuskomennot eivät ole välttämättömiä, mutta jos olet käyttänyt lopetuskomentoa jossain kohdassa, sinun tulee käyttää niitä jokaisen rivin ja solun yhteydessä. Muuten taulukkosi tulee näkyviin puutteellisena.

- Ovatko kaikki taulukon sisällä olevat tekstit ja kuvat solukomentojen sisällä? Jos sijoitat taulukkoosi tekstiä tai kuvia siten, että ne eivät ole solukomentojen sisällä, ne tulevatkin näkyviin taulukon ulkopuolelle. Esimerkki väärin toteutetusta taulukosta:

```
<tr><p>Tämä teksti tulee väärään paikkaan!
<td>Solu</td><td>Toinen solu</td>
</tr>
```

- Vastaavatko aloitus- ja lopetuskomennot toisiaan? Taulukkoa kirjoitettaessa käytetään paljon HTML-komentoja ja joskus kirjoittaessa tulee virheitä. Kun suunnittelet ja teet taulukkoja, kannattaakin käyttää hyväkseen sitä, että HTML-koodin sisään voi sijoittaa välilyöntejä haluamansa määrän. Tällä tavalla voit sisentää taulukon rivin esimerkiksi yhdellä välilyönnillä ja kunkin solun kahdella välilyönnillä. Erittäin hyödyllistä tämä on, jos teet solujen sisään uusia taulukoita. Muista käyttää ahkerasti myös kommentteja. Alla on esimerkki monimutkaisesta taulukosta:

```
<table>
 <tr>
  <!-- eka rivi, kaksi solua -->
  <td>Foo</td>
  <td>Bar</td>
 </tr>
 <tr>
  <td>
  <!-- toka rivi, kaksi solua -->
  <!-- toka rivi, ensimmäinen solu -->
   <table>
    <tr>
     <td>Sisäkkäinen taulukko!</td>
```

```
      <td>Upeeta!</td>
     </tr>
    </table>
   </td>
   <!-- tokan rivin toinen solu -->
   <td>Baz</td>
  </tr>
 </table>
```

ERIKOISKAPPALEET

HTML sisältää kokonaisia kappaletyyppejä suuremmille asiakokonaisuuksille. Nämä kappaleet muistuttavat toiminnaltaan tavallisia tekstikappaleita. Kun käytät erikoiskappalekomentoja, niihin ei tarvitse lisätä tekstikappaleen aloituskomentoa **<p>**.

ESIMUOTOILTU TEKSTI

Esimuotoiltua tekstiä käytetään silloin, kun on tärkeää että teksti tulee näkyviin lukijan ruudulle täsmälleen samalla tavalla kuin HTML-tiedostoon on kirjoitettu. Esimuotoiltu teksti tulee näkyviin ilman muutoksia - sen välilyönnit ja rivinvaihtomerkit tulevat näkyviin muuttumattomina. Tämä tyyli onkin nopein tapa muuttaa jo tietokoneella olevaa tekstiä WWW-sivuksi: kun merkitset koko tekstin esimuotoilluksi tyyliksi, sinun ei tarvitse lisätä itse tekstiin HTML-komentoja.

Esimuotoiltu tekstikappale sijoitetaan **<pre>**- ja **</pre>**-komentojen väliin. Esimuotoiltu teksti tulee näkyviin lukijan ruudulle tasalevyisellä kirjasintyypillä (esimerkiksi **Courier**). Koska esimuotoillussa tyylissä merkitään rivinloput ja kappaleiden välit jo kirjoitusvaiheessa, siinä ei tarvita **<p>**- ja **
**-komentoja. Voit lisätä esimuotoiltuun tyyliin normaaleja linkkejä ja tekstimuotoiluja (esimerkiksi lihavointia). Kappalemuotoiluja (otsikot, osoitekappaleet ja niin edelleen) ei tulisi käyttää esimuotoillun tekstin sisällä.

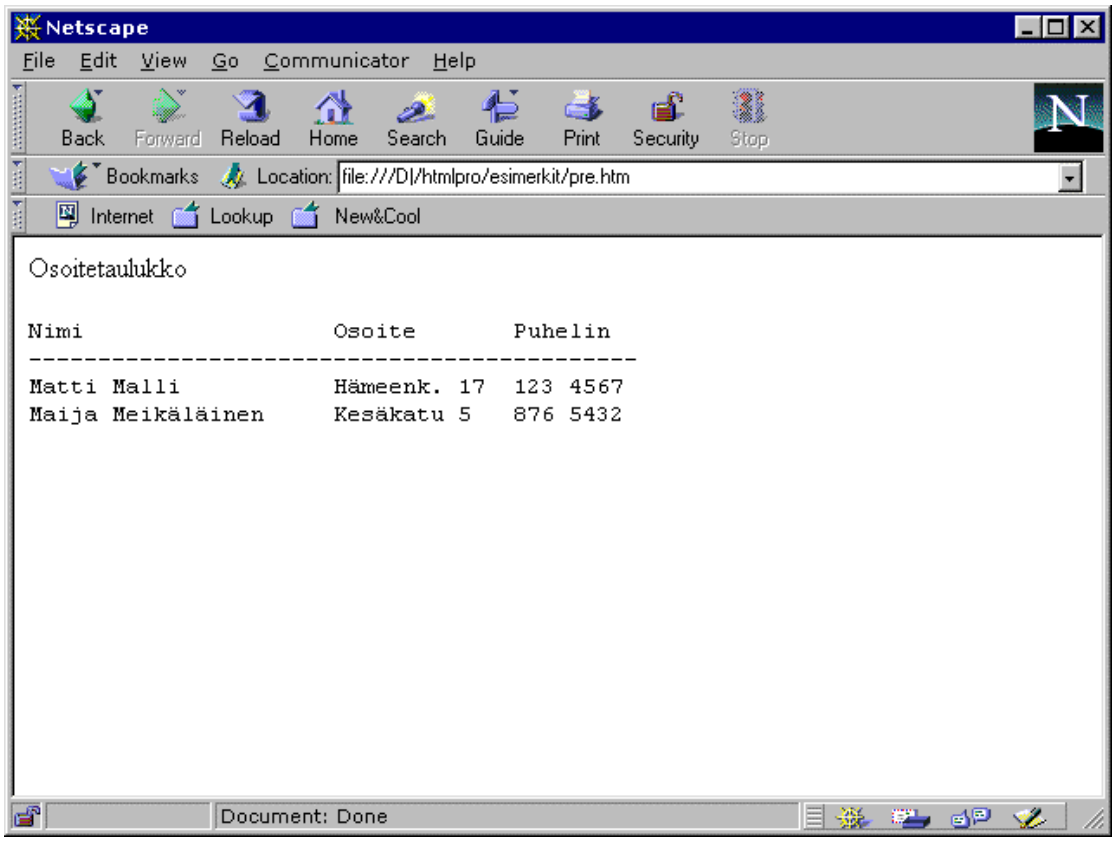
Koska merkkejä **<**, **>** ja **&** käytetään HTML:ssä erikoistarkoituksiin, sinun täytyy kuitenkin korvata nämä merkit niitä vastaavilla entiteeteillä. Tabulaattorien käyttö on mahdollista mutta ei suositeltavaa, sillä ne saattavat toimia eri ohjelmissa eri tavoilla.

Esimuotoiltua tyyliä voi käyttää esimerkiksi yksinkertaisten taulukoiden toteuttamiseen, ohjelmalistauksien tai ohjelmien tulosteiden esittämiseen, sisentämiseen ja yksinkertaiseen palstoittamiseen.

Esimerkki taulukon toteuttamisesta esimuotoillulla tyylillä:

```
<p>Osoitetaulukko</p>
<pre>
Nimi                    Osoite        Puhelin
-------------------------------------------
Matti Malli             Hämeenk. 17   123 4567
Maija Meikäläinen       Kesäkatu 5    876 5432
</pre>
```

Lukijan ruudulla näkyvä teksti näyttää samalta kuin yllä olevassa esimerkissä, vain **<pre>**-komennot puuttuvat:



Huomaa, kuinka rivinvaihtoja ei tarvinnut merkitä erikseen. **<pre>** ja **</pre>** -komentojen väliin sijoitetusta tekstistä otetaan mukaan kaikki välilyönnit ja rivinvaihtomerkit.

LISTAT

HTML tarjoaa mahdollisuuden näyttää kuvaruudulla tekstiä listoina. Listat voivat olla numeroituja, ne voivat sisältää ranskalaisia viivoja, tai ne voivat olla määritelmälistoja.

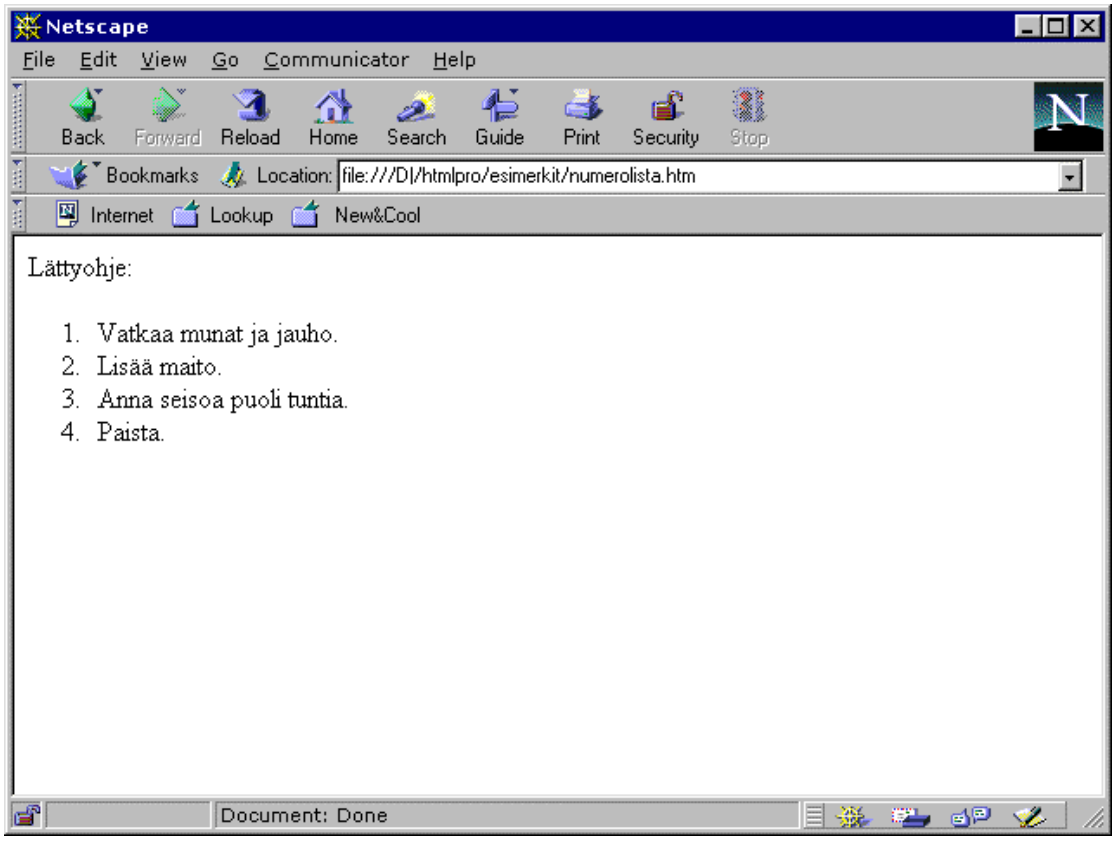
Numeroituja listoja (**Ordered list**) tehtäessä kirjoitetaan ainoastaan listan eri kohdat. Selain hoitelee automaattisesti listan numeroinnin.

Numeroitu lista aloitetaan komennolla **
** ja lopetetaan komennolla **
**. Näiden kahden komennon väliin sijoitetaan listan kohdat. Jokainen kohta merkitään komennolla **
**. Listan kohdan loppuun voi rakenteen selkeyttämiseksi lisätä lopetuskomennon **
**. Se ei ole kuitenkaan välttämätön.

Esimerkiksi ruokaresepti voitaisiin kirjoittaa numeroituna listana:

```
<p>Lättyohje:
<ol>
<li>Vatkaa munat ja jauho.</li>
<li>Lisää maito.</li>
<li>Anna seisoa puoli tuntia.</li>
<li>Paista.</li>
</ol>
```

Lukijalle näkyvänä tuloksena on numeroitu lista:



Listan kohta voi olla kuvaruudun leveyttä pitempi: selain osaa automaattisesti muotoilla sen ruudulle oikein sisennettynä.

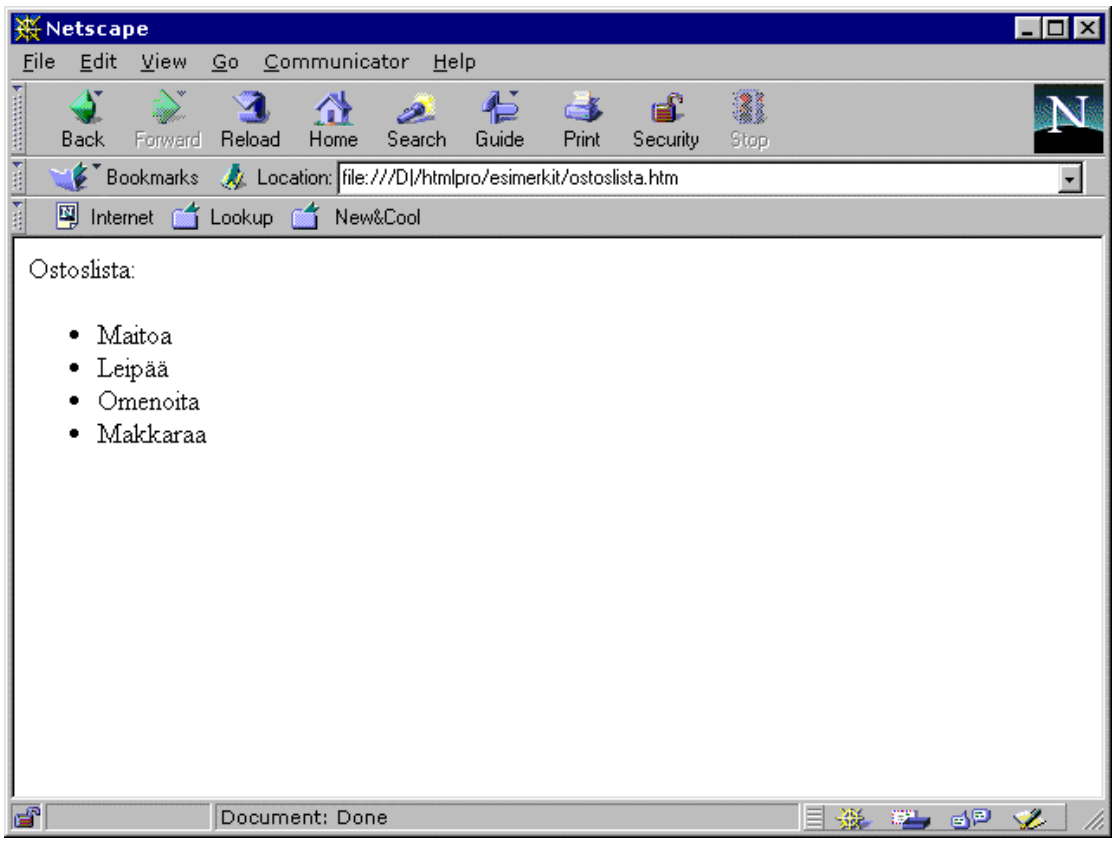
Ranskalaiset viivat (**Unordered list**) toimivat samalla tavalla kuin numeroidut listat, mutta ilman numeroita. Näissä listoissa jokainen listan kohta merkitään ranskalaisella viivalla, tai pallolla, kuten useimmissa selaimissa.

Lista sijoitetaan komentojen **
** ja **
** väliin. Jokainen listan kohta merkitään komennolla **
**, aivan kuten numeroiduissa listoissakin. Listan kohdan voi lopettaa komennolla **
** (tämä lopetuskomento ei ole välttämätön).

Esimerkki ranskalaisista viivoista:

```
<p>Ostoslista:
<ul>
<li>Maitoa
<li>Leipää
<li>Omenoita
<li>Makkaraa
</ul>
```

Tässä kuvaruudulla näkyvä valmis lista sellaisena kuin se näkyy selaimessa:



Määritelmälistat (**Definition list**) koostuvat termeistä sekä niiden määrittelyistä. Lista aloitetaan komennolla **<dl>** ja lopetetaan komentoon **</dl>**. Näiden väliin jäävälle alueelle sijoitetaan termi-määritelmä -pareja. Määriteltävä termi merkitään komennolla **<dt>** ja sen määritelmä komennolla **<dd>**. Termi- ja määrittelyriveille voi lisätä halutessaan sivun rakennetta selkiyttämään lopetuskomennot **</dt>** ja **</dd>**.

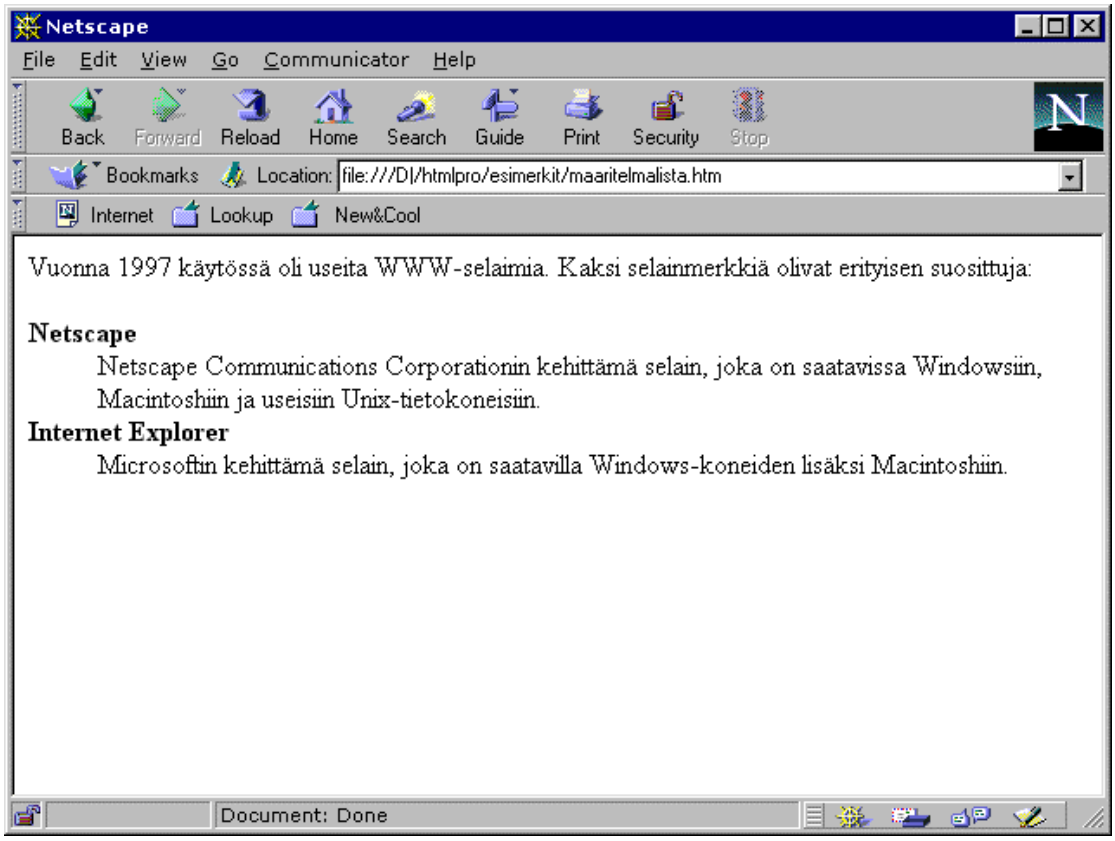
Lukijalle näkyvässä lopputuloksessa termit näytetään useimmiten normaalina tekstinä ja määritelmät sisennettyinä kappaleina, niin ikään normaalina tekstinä.

Esimerkki määritelmälistasta:

```
<p>Vuonna 1997 käytössä oli useita WWW-selaimia. Kaksi
selainmerkkiä olivat erityisen suosittuja:
<dl>
<dt><b>Netscape</b></dt>
<dd>Netscape Communications Corporationin kehittämä
```

```
selain, joka on saatavissa Windowsiin, Macintoshiin ja
useisiin Unix-tietokoneisiin.</dd>
<dt><b>Internet Explorer</b></dt>
<dd>Microsoftin kehittämä selain, joka on saatavilla
Windows-koneiden lisäksi Macintoshiin.</dd>
</dl>
```

Lopputulos:



Termien ja määritelmien tulisi esiintyä listassa aina pareittain. Yhtä määrittelyä kohti voi kuitenkin olla monta termiä.

OSOITEKAPPALEET

Osoitekappaleet tehdään aloittamalla kappale komennolla **<address>** ja päättämällä se komentoon **</address>**. Tällä kappaletyypillä voit esittää erilaisia osoitteita. Yleinen tapa on merkitä sivun loppuun tekijän nimi ja sähköpostiosoite osoitetyylillä.

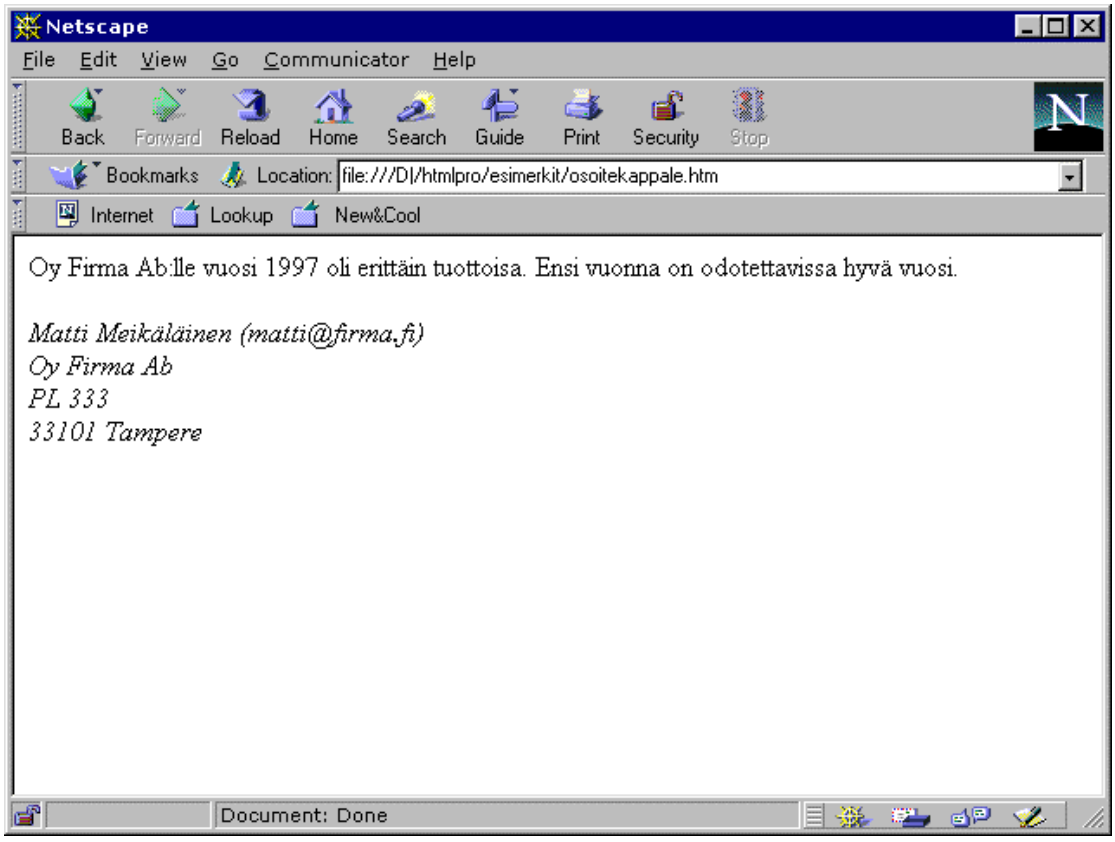
Aivan kuten leipätekstikappaleissakin, kirjoittamassasi koodissa olevat rivinvaihdot jäävät huomioimatta. Jos haluat tehdä esimerkiksi useamman rivin mittaisia osoitteita (esimerkiksi nimi, osoite ja postitoimipaikka), mer-

kitse jokaisen rivin loppu rivinvaihtokomennolla **
**. Voit tietysti myös sijoittaa peräkkäin useita erillisiä osoitekappaleita.

Esimerkkejä osoitteista:

```
<p>Oy Firma Ab:lle vuosi 1997 oli erittäin tuottoisa.
Ensi vuonna on odotettavissa hyvä vuosi.</p>
<address>
Matti Meikäläinen (matti@firma.fi)<br>
Oy Firma Ab<br>
PL 333<br>
33101 Tampere
</address>
```

Osoitteet tulostuvat selaimissa sen määrittelyistä riippuen. Useimmat selaimet muotoilevat osoitteen kursiivilla muotoilluksi erilliseksi kappaleeksi:



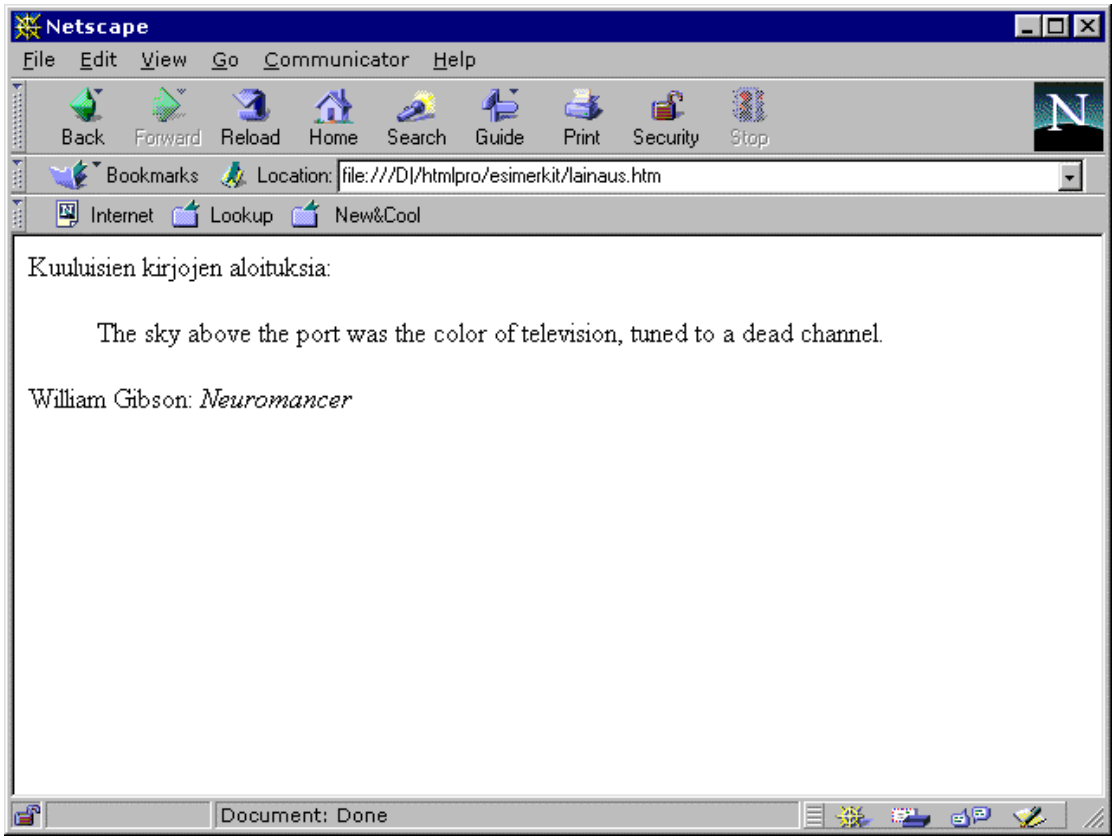
LAINAUKSET

Jos lainaat osia muista teksteistä, voit sijoittaa ne tällä komentojen **<blockquote>** ja **</blockquote>** väliin. Tällä tavalla selain osaa muotoilla

lainaukset muusta tekstistä paremmin erottuviksi. Nykyiset ohjelmat näyttävät lainaukset yleensä samalla tekstityylillä kuin leipätekstinkin, mutta sisennettyinä. Esimerkki:

```
<p>Kuuluisien kirjojen aloituksia:
<blockquote>
The sky above the port was the color of television,
tuned to a dead channel.[3]
</blockquote>
<p>William Gibson: <cite>Neuromancer</cite>
```

Sama teksti lukijalla näkyy näin:



[3]William Gibson, *Neuromancer*, 1984, s. 9.

KUVANKÄSITTELY

KUVAFORMAATIT

Kuten aikaisemmissa luvuissa kerrottiin, WWW-sivuilla voi käyttää kuvia joko **gif**- tai **jpeg**-muodossa. Tulevaisuudessa jotkin muutkin formaatit saattavat olla mahdollisia. Kuvaformaateilla on jonkin verran eroja.

GIF

Gif on yleisin WWW-sivuilla käytetty kuvaformaatti. **Gif**-kuvat pakataan tallennettaessa siten, että kuvissa olevat laajat väripinnat pakataan pienempään kokoon. Tämän ansiosta kuvat vievät sekä vähemmän levytilaa että siirtyvät verkon yli nopeammin. Kuvien pakkaus ei hävitä kuvien yksityiskohtia toisin kuin **jpeg**-formaatissa.

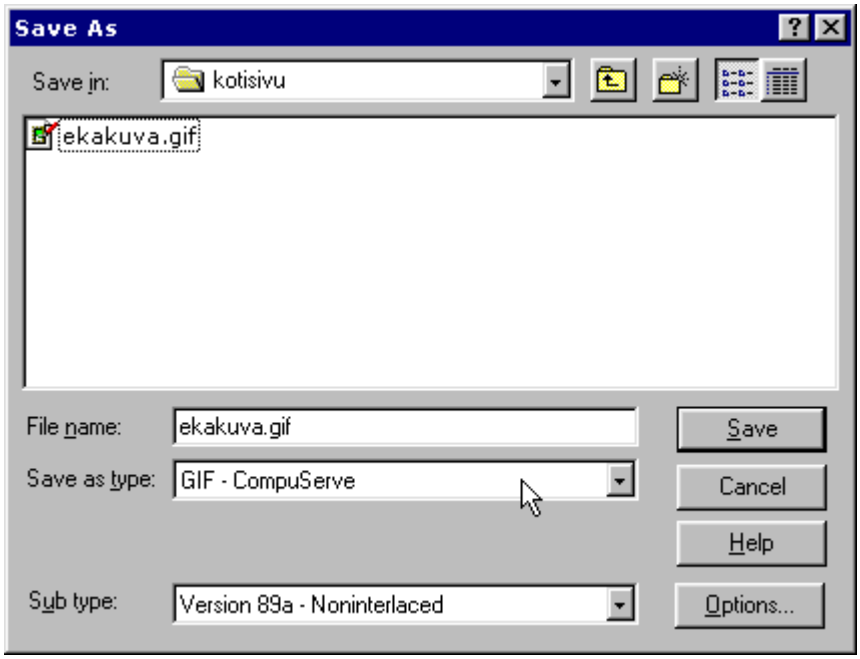
Gif-kuvat voivat sisältää maksimissaan 256 yhtäaikaista väriä. Kuvat joissa käytetään suurempia värimääriä (esimerkiksi skannatut valokuvat) joudutaan käsittelemään siten, että kuvan värimäärä ei ylitä tätä. **Gif**-kuvat sopivatkin parhaiten erilaisten piirroksien ja kaaviokuvien esittämiseen.

Jos kuvassa ei tarvita edes 256 yhtäaikaista väriä, **gif**-kuvien värimäärän voi tiputtaa myös pienemmäksi. Tämä pienentää kuvan viemää levytilaa ja nopeuttaa sen siirtämistä verkon yli. Esimerkiksi suurin osa WWW-sivuilla olevista mainoskuvista on käsitelty 16-värisiksi, jotta kuvat latautuisivat mahdollisimman nopeasti.

Gif-kuvat sisältävät jonkin verran erikoistoimintoja, jotka tekevät niistä WWW-käyttöön soveliaita. Kuvista voi tehdä animoituja, eli niihin voi liittää monta kuvaa, jotka peräkkäin näytettyinä saavat kuviin liikettä tai erilaisia efektejä.

Gif-kuviin voi liittää myös "läpinäkyvän" värin, jolloin epäsäännöllisten kuvioiden liittäminen taustakuvan päälle tulee paljon helpommaksi.

Kun haluat tallentaa kuvan **gif**-muodossa **Paint Shop Pro** -ohjelmassa, valitse tallennusmuodoksi "**GIF - CompuServe**". **Gif**-kuvien tiedostonnimi päättyy aina tekstiin **`.gif`**, eli esimerkiksi **`ekakuva.gif`**.



Gif-formaatilla on kaksi alatyyppiä: 87a ja 89a. Uudempi versioista, 89a, mahdollistaa animaatioiden ja muiden erikoisefektien toteuttamisen, joten käytä sitä aina tallentaessasi **gif**-kuvia.

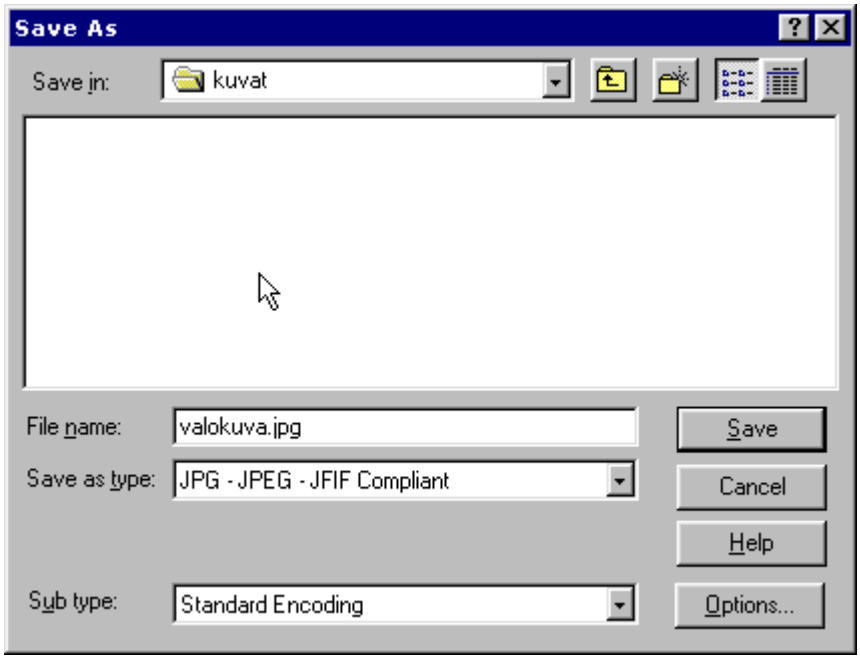
JPEG

Jpeg on voimakkaasti pakkaava kuvaformaatti. Tämä tarkoittaa sitä, että tallennettaessa kuvista poistetaan pienimpiä yksityiskohtia, jolloin kuvatiedostot pakkautuvat todella pieneen tilaan. Tällä tavalla esimerkiksi valokuvista, jotka **gif**-muodossa olisivat suurikokoisia ja joiden lataus kestäisi kauan, saadaan pieniä ja nopeasti latautuvia. Suurikokoistenkin kuvien WWW-sivuille sijoittaminen on järkevää **jpeg**-formaattia käytettäessä.

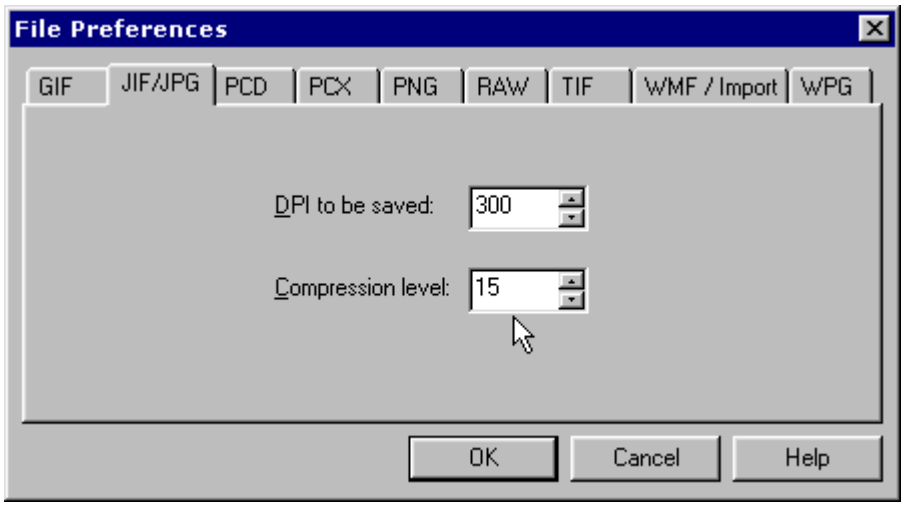
Jpeg-kuvat sisältävät aina automaattisesti 24-bittisen värikartan, eli kuvassa voidaan käyttää miljoonia värejä. **Jpeg**-muodossa kuvat voidaan siis tallentaa aina niin värikkäinä kuin vain mahdollista. Tämän ansiosta **jpeg** sopii erityisesti valokuvien ja muiden värikkäiden kuvien tallentamiseen.

Jpeg-kuvissa ei voida käyttää **gif**-kuvissa mahdollisia erikoisefektejä: niissä ei voi olla läpinäkyvää väriä eikä niiden avulla voi tehdä animaatioita.

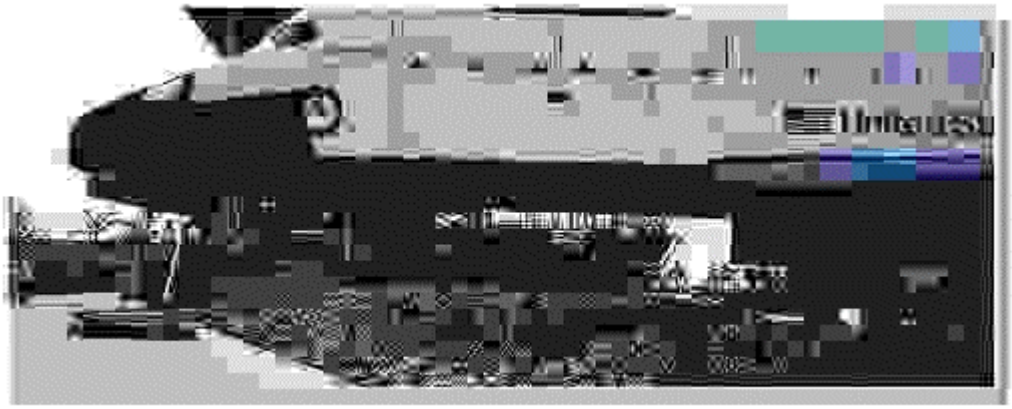
Kun tallennat kuvia **jpeg**-muotoon **Paint Shop Pro:**lla, valitse tallennustyypiksi "**JPG - JPEG - JFIF Compliant**". **Jpeg**-kuvien tarkennin on **`.jpg`**, eli esimerkiksi **`valokuva.jpg`**.



Jpeg-kuvaa tallennettaessa valitaan myös sen pakkausmäärä. Tämä tapahtuu napsauttamalla tallennusikkunasta **Options**-painiketta. Saat ruudullesi ikkunan, jossa voit määritellä **jpeg**-kuvan pakkauksen määrän:



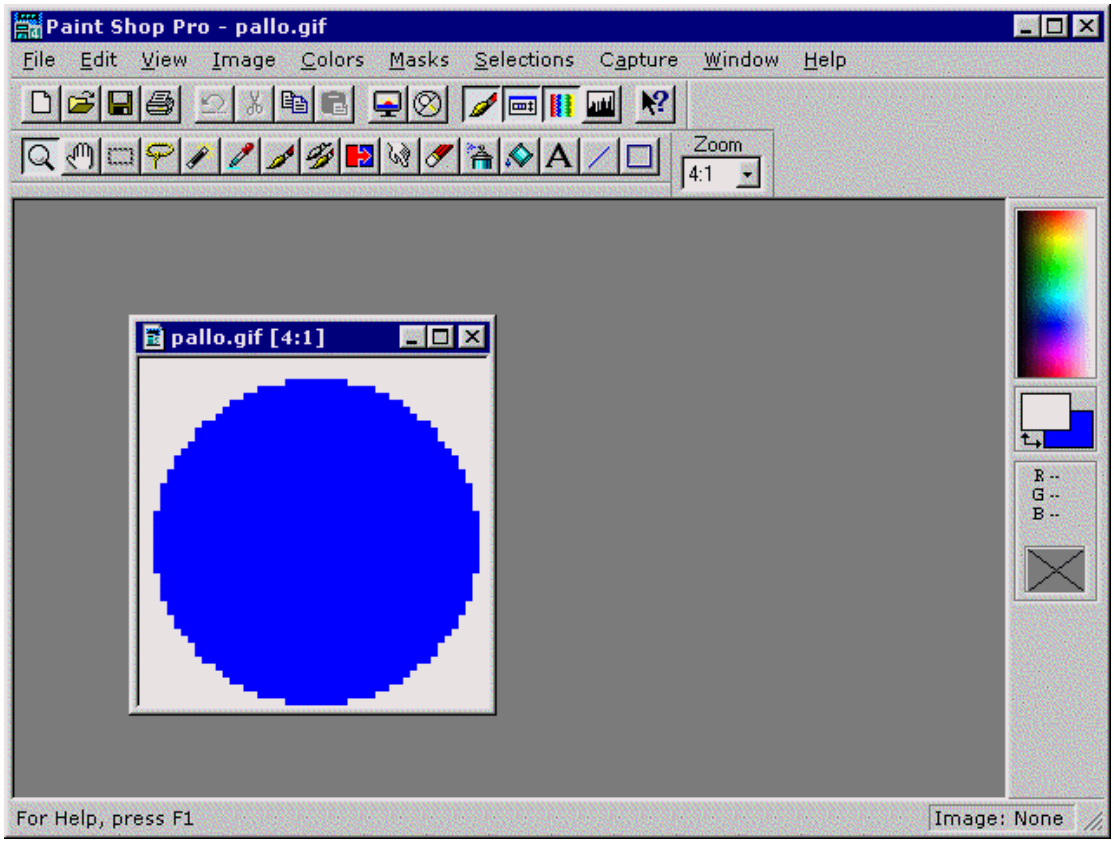
Kohdassa **Compression level** määritellään pakkauksen voimakkuus. Mitä suurempi numero, sitä pienempään tilaan kuva pakkautuu, mutta sitä enemmän siitä myös katoaa yksityiskohtia ja värejä. Maksimiteholla (99) pakatussa **jpeg**-kuvassa suurin osa väreistä ja yksityiskohdista katoaa:



Parhaat tallennustulokset saat aikaan vain kokeilemalla, mutta jo pakkausteholla 15 kuvat pakkaantuvat hyvin pieneen tilaan.

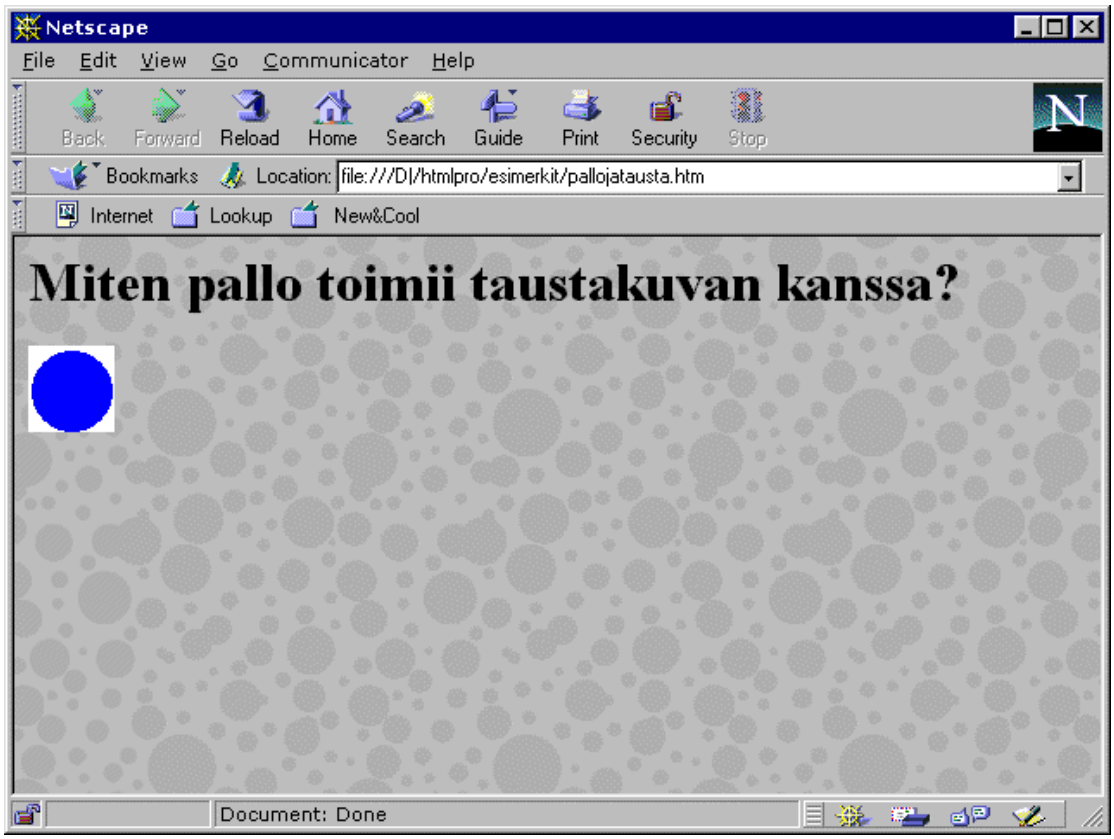
LÄPINÄKYVÄT VÄRIT

Kaikki WWW-sivuilla käytettävät kuvat eivät ole välttämättä nelikulmion muotoisia. Otetaan esimerkiksi kuva sinisestä pallosta valkoisella taustalla (kuvassa suurennettuna **Paint Shop Pro**n ruudulla):



Kun käytät tätä kuvaa WWW-sivulla, se sujuu ongelmitta niin kauan kuin sivullasi on valkoinen taustaväri. Jos muutat taustavärin joksikin toiseksi, joudut muokkaamaan kuvan taustaväriä vastaavasti.

Törmäät kuitenkin ongelmiin, jos haluat lisätä tämän kuvan sellaiselle sivulle, jossa on käytetty taustakuvaa:



Vaikka muokkaisit pallon taustavärin täysin taustakuvaan sopivaksi, tausta menisi pilalle pallon siirtyessä eri paikkaan ruudulla. Tarvitaan siis toinen ratkaisu.

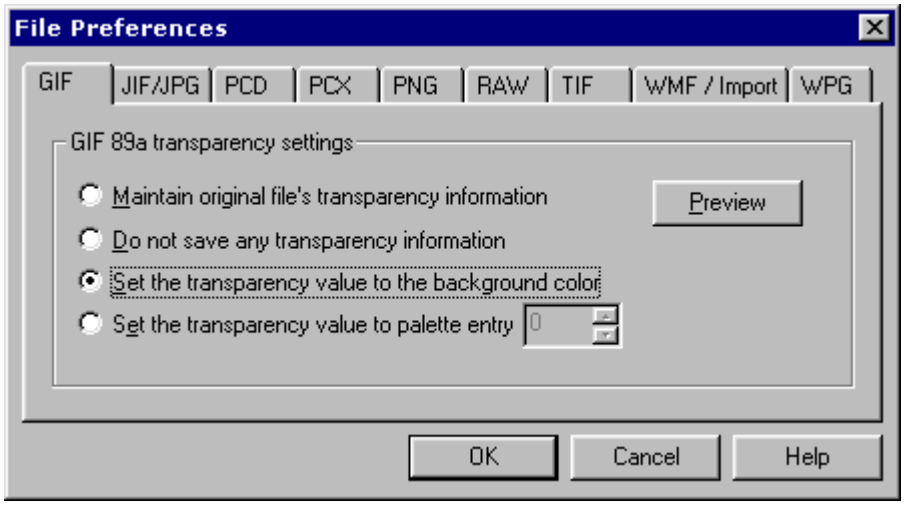
Ongelman ratkaisee läpinäkyvä taustaväri. Pallon taustalla oleva valkoinen väri voidaan määritellä läpinäkyväksi, jolloin taustakuva näkyy siitä läpi. Määrittely tapahtuu **Paint Shop Pro:ssa**.

Kun teet läpinäkyvää väriä kuvaan, yksi kuvan väreistä muutetaan läpinäkyväksi. Varmistu ensin, että samaa väriä ei käytetä jossain kuvassa (koska silloin WWW-sivun tausta näkyisi läpi myös kuvan keskeltä). Esimerkin pallossa läpinäkyväksi muutettava väri on valkoinen eikä sitä ole käytetty pallon sisällä.

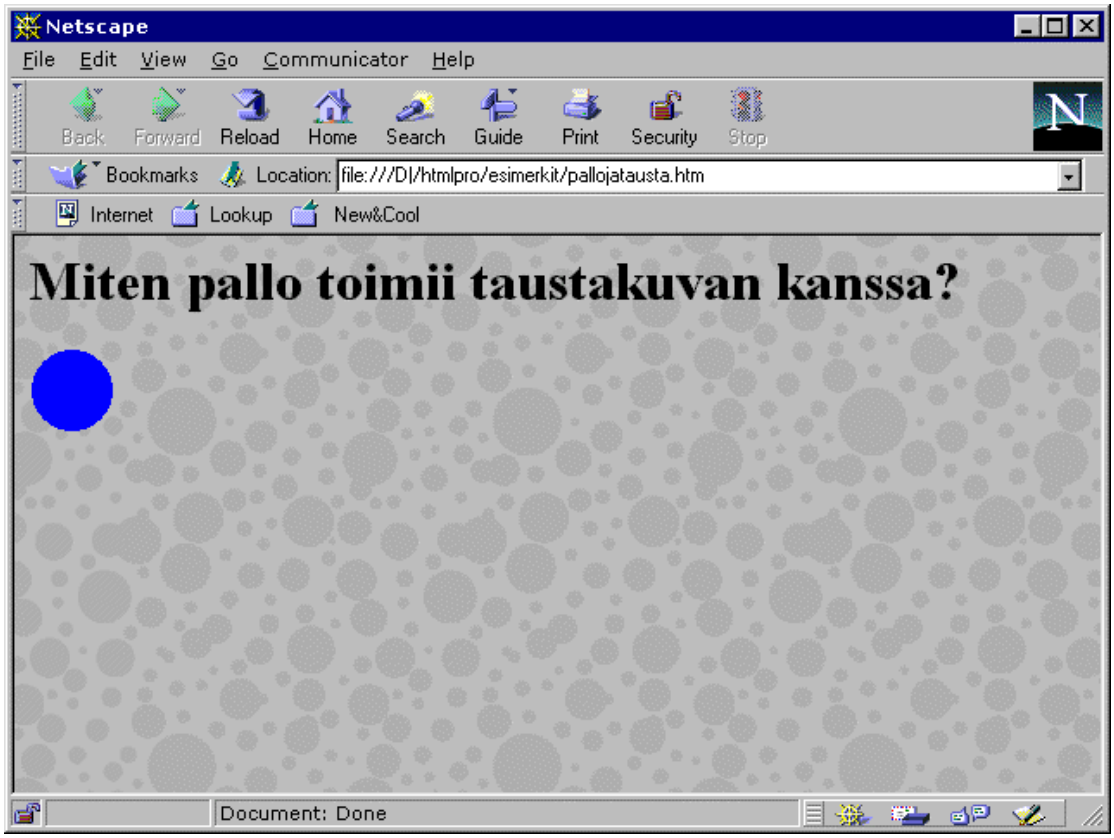
Etsi seuraavaksi **Dropper**-työkalu **Paint Shop Pron** valikoista (kuvassa alas painettu painike):

Napsauta tämän jälkeen hiiren oikeanpuoleisella napilla väriä, jonka haluat muuttaa läpinäkyväksi. Valitsemasi väri tulee näkyviin **Paint Shop Pron** värivalikkoon aktiiviseksi taustaväriksi.

Tämän jälkeen voit avata tallennusikkunan. Valitse tallennusformaatiksi "**GIF - Compuserve**" kuten ennenkin **gif**-kuvia tallentaessasi. Ennen tallentamista napsauta kuitenkin **Options**-painiketta. Saat näkyvillesi **gif**-kuvan tallennukseen liittyvien lisätoimintojen valikon:



Valitse valikosta toiminto "**`Set the transparency value to the background color`**" ja napsauta "OK". Kun nyt tallennat kuvan, valitsemasi valkoinen väri muuttuu läpinäkyväksi. Tämän jälkeen kuvasi toimii oikein myös taustakuvalla varustetuilla sivuilla:

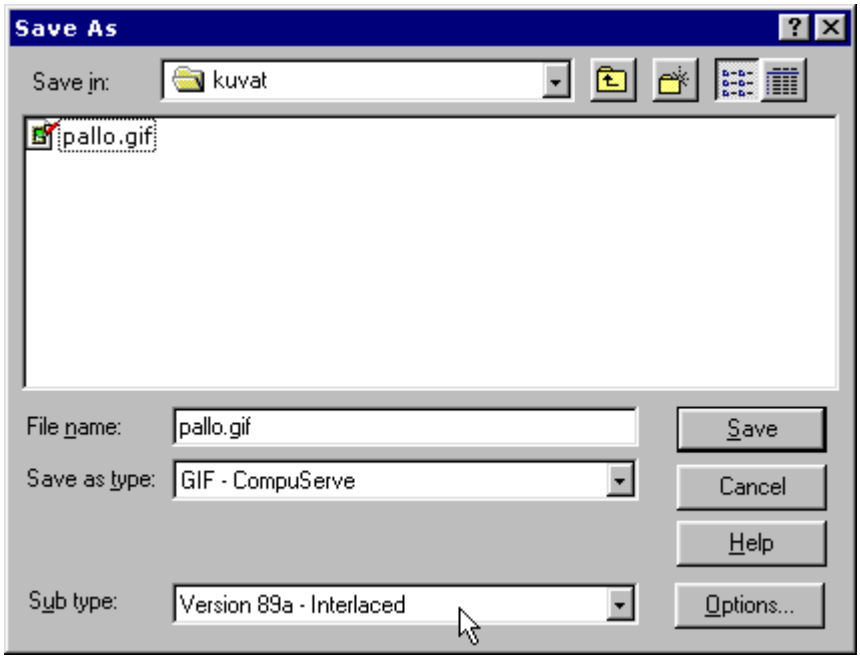


VÄHITELLEN NÄKYVIIN TULEVA KUVA

Etenkin isokokoisten kuvien lataus saattaa kestää modeemikäyttäjillä jonkin aikaa. Kuvat on mahdollista tallentaa siten, että niistä tulee näkyviin ensin huonompilaatuinen vedos, joka tarkentuu hiljalleen käyttäjän näytöllä. Se ei nopeuta kuvan latausta, mutta käyttäjä saa nopeammin katsottavaa odotellessaan sivun latautumista.

INTERLACE

Gif-kuvissa vähitellen näkyviin tuleminen toteutetaan tallentamalla kuva **interlace**-muodossa. Tämä vaihtoehto valitaan **Paint Shop Pron** tallennusikkunasta kohdasta **Sub type**:



PROGRESSIVE JPEG

Jpeg-kuvassa vastaava toiminto saadaan aikaan tallentamalla kuva **progressive**-muodossa. Kun olet valinnut kuvan tallennusmuodoksi **jpegin**, kohdassa **Sub type** on valittavana kohta "**Progressive Encoding**".

VÄRIMÄÄRÄT

Kuvien värimäärä vaikuttaa suoraan sekä kuvan tiedostokokoon että siihen, miten nopeasti kuva siirtyy lukijan näytölle. Tällä on merkitystä erityisesti silloin, kun katsojalla on käytössään modeemi.

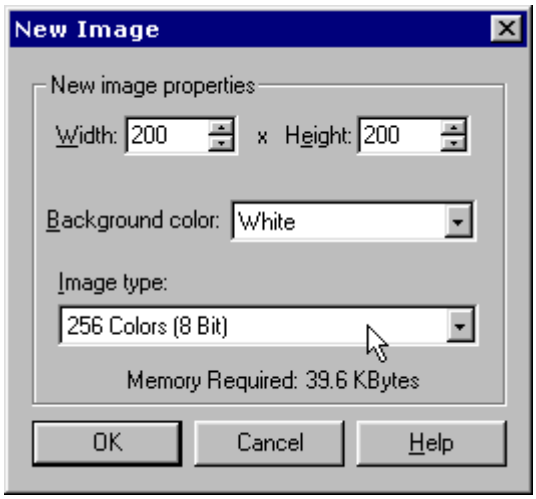
Jpeg-kuvat tallentuvat aina maksimivärimäärällä, joten niissä kuvan värimäärällä ei ole merkitystä. **Gif**-kuvissa sen sijaan kuvan värimäärän käsittelyllä voidaan kuvan kokoa kutistaa merkittävästi.

Gif-kuvat voidaan tallentaa eri bittisyvyyksiä käyttäen. Bittisyvyys kuvaa sitä, kuinka monta bittiä tarvitaan yhden kuvassa olevan pikselin esittämiseen. Maksimissaan **gif**-kuvassa yksi piste voi viedä 8 bittiä, jolloin kuvassa voi olla siis 2^8, eli 256 väriä. Bittisyvyyttä voi pudottaa **Paint Shop Prossa** neljään (16-värinen kuva) tai yhteen (2-värinen kuva).

16-värinen kuva vie levytilaa ja siirtoaikaa tämän perusteella siis vain puolet 256-värisen kuvan tarpeesta. Kaksivärinen kuva siirtyy kahdeksan kertaa nopeammin. Käytännössä erot tasoittuvat hieman jokaiselle kuvalle

yhteisten tietojen vaikutuksesta, mutta kuvien värimäärän pudottaminen on silti kannattavaa.

Paint Shop Prota käyttäessäsi voit valita kuvan värimäärän jo uutta kuvaa tehdessäsi. Uutta kuvaa aloittaessasi sinulta kysytään haluamaasi värimäärää:



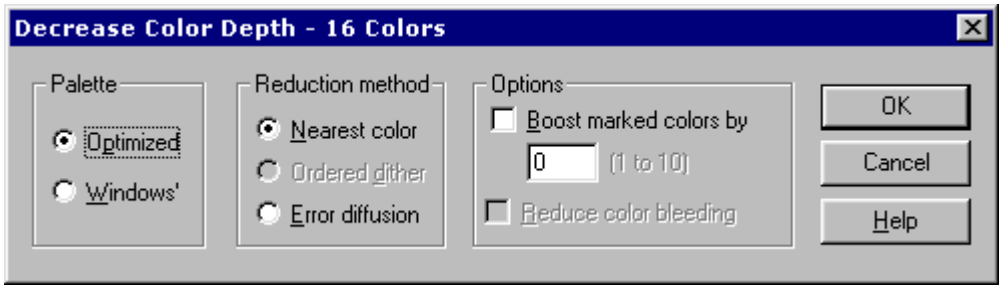
Värimäärää vaihtaessasi näet kuvan alapuolelta, miten paljon tilaa kuva tulee viemään. Vaihtamalla värimäärän 256-värisestä 16-väriseksi huomaat, että tilantarve vaihtuu 39 kilotavusta 19 kilotavuun silloin, kun kuvan koko on 200 x 200 pikseliä.

Modeemilla, jonka nopeus on 28 800 bittiä sekunnissa (n. 2800 tavua sekunnissa) 256-värisen kuvan siirtäminen kestäisi noin 14 sekuntia ja 16-värisen noin 7 sekuntia.

Nämä arvot eivät tosin vastaa käytännön olosuhteita, sillä **gif**-kuvahan pakataan tallennettaessa. Kuvan lopullinen kutistuminen riippuu kuvassa olevien yksityiskohtien määrästä - mitä enemmän laajoja väripintoja, sitä pienempään tilaan kuva pakkautuu.

Kuvaa käsitellessäsi voit halutessasi lisätä ja vähentää värejä. Voit valita **Colors**-valikosta toiminnot **Increase Color Depth** (lisää värimäärää) tai **Decrease Color Depth**. Näiden avulla voit kokeilla, miten värimäärän muutokset vaikuttavat kuvaasi. Voit aina perua muutoksen **Edit**-valikon toiminnolla **Undo**.

Kun pudotat kuvan värimäärää, **Paint Shop Pro** kysyy sinulta menetelmää, joilla puuttuvat värit korvataan:



Voit valita värien rasteroinnin (**dither**, **diffusion**) tai korvata poistuvat värit kuvaan jäävillä väreillä.

Edistyneemmissä kuvankäsittelyohjelmissa kuvan värimäärää voi kontrolloida vielä tarkemmin antamalla jopa haluttujen värien lukumäärän.

GIF-ANIMAATIO

Gif-kuvien avulla on mahdollista tehdä myös animaatioita. Ne näyttävät tavallisilta kuvilta ja niitä käsitelläänkin samoilla komennoilla - ainoana erona on se, että kuvat liikkuvat ruudulla.

Animaatiot ovat mahdollisia **gif**-formaattiin tehdyn lisäyksen ansiosta. Yhden **gif**-kuvan sisään on mahdollista upottaa useampia kuvia ja määritellä kuvat näytettäviksi peräjälkeen. Näin syntyy yksinkertainen animaatio. Kuvien vaihtumiselle on mahdollista määritellä eripituisia viiveitä: **gif**-animaatio voi sisältää 3 kuvaa, joita näytetään peräkkäin vaikkapa sekunnin välein. Lisäksi on mahdollista määritellä toisto: kolmen kuvan animaatio voidaan määritellä ikuiseksi silmukaksi, jolloin kuvia pyöritetään jatkuvasti (1, 2, 3, 1, 2, 3 ja niin edelleen). Toistolle voidaan antaa jokin arvo, jolloin kuvasarjaa toistetaan vain esimerkiksi 10 kertaa.

Gif-animaatioita on mahdollista tehdä tarkoitusta varten tehdyillä ohjelmilla. Nämä ohjelmat ovat yleensä hyvin yksinkertaisia eivätkä ne itsessään sisällä minkäänlaisia kuvankäsittelytoimintoja. Sen sijaan animaation yksittäiset kuvat piirretään jollain muulla ohjelmalla (tavalliset kuvankäsittelyohjelmat kelpaavat tähän tarkoitukseen mainiosti) ja vasta lopuksi yhdistetään animaatioksi erillisellä ohjelmalla.

Yksi tällainen ohjelma on **Gif Construction Set**. Se on saatavissa osoitteesta:

http://www.mindworkshop.com/alchemy/gifcon.html

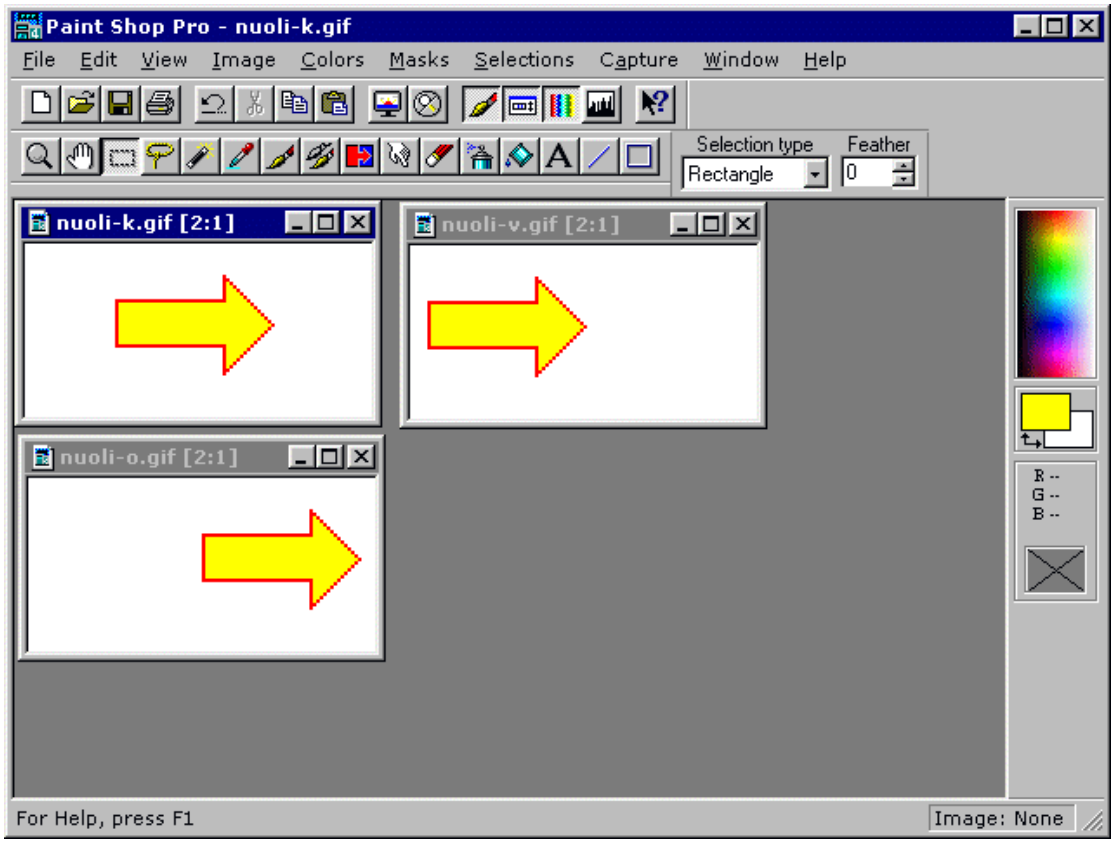
ANIMAATION TEKEMINEN

Gif-animaation tekeminen lähtee liikkeelle yksittäisten kuvien piirtämisestä. Liikkuvia kuvia tehtäessä jokainen kuva on hieman erilainen kuin edellinen. Kun kaikki kuvat näytetään nopeasti peräkkäin, katsojalle syntyy illuusio liikkeestä.

Kun teet WWW-sivuilla katsottavaksi tarkoitettuja animaatioita, sinun on otettava huomioon samat seikat kuin muitakin kuvia tehdessä: kuvien koko (sekä kuvaruudulla näkyvä koko että niiden tiedostokoko) ja värimäärä. **Gif**-animaatioita tehdessä täytyy kiinnittää erityisesti huomiota tiedostokokoon. Kaikki animaatioon kuuluvat yksittäiset kuvat upotetaan saman tiedoston sisään. Sen vuoksi tiedoston koko kasvaa hurjasti, etenkin jos kuvat ovat isoja ja animaatioon kuuluu monta kuvaa. **Gif**-animaation tiedostokoko on siihen käytettyjen yksittäisten kuvien kokojen summa - vaikka kaikki animaation kuvat olisivatkin samankaltaisia.

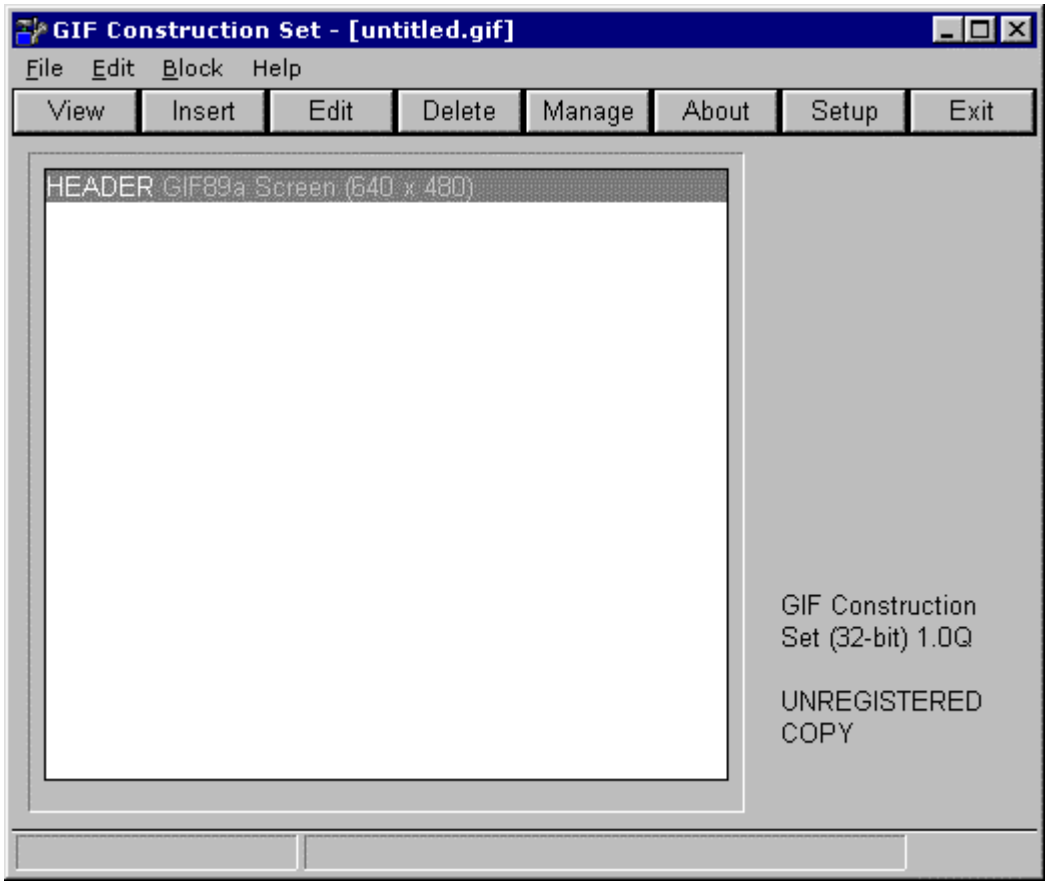
Verkossa käytettävät animaatiot sisältävätkin usein vain muutamia kuvia, jotta niiden lataus ei kestäisi kauan. Tällöin itse animaation toteuttamisessa tulee tehdä kompromisseja: sulavat liikkeet ja pitkät animaatiot eivät ole käytännöllisiä. Monet animaatiot sisältävät joko karkeaa liikettä tai peräti toisistaan eroavia kuvia, joita näytetään kuvaruudulla verkkaiseen tahtiin. Mitä pienempi animaation kuvaruudulla näkyvä koko on, sitä enemmän siihen saa mahdutettua yksittäisiä kuvia. Animaation lataamisen ei tulisi kestää juurikaan tavallisen kuvan latausta kauemmin: noin 50 kilotavun kokoinen animaatio on jo iso.

Tehdään esimerkissä animaatio vaakasuunnassa heiluvasta nuolesta. Animaatioon tulee kolme kuvaa: nuoli vasemmalla, nuoli keskellä ja nuoli oikealla. Kun näitä kuvia näytetään peräkkäin järjestyksessä vasen-keski-oikea-keski, saadaan aikaan animaatio, jossa nuoli näyttää heiluvan sivuttain. Animaation kuvat piirretään tavallisella kuvankäsittelyohjelmalla ja tallennetaan levylle **gif-**muodossa. Kuvat näyttävät tältä:

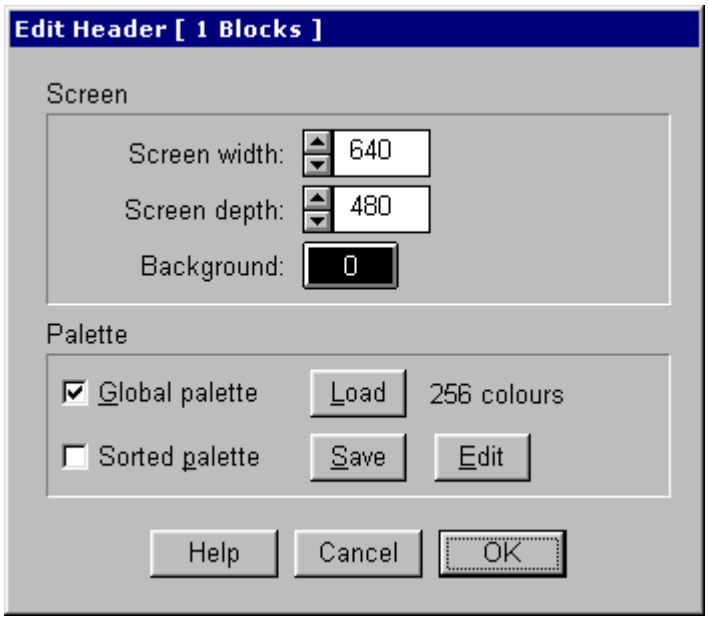


Jokainen kuva tallennetaan omaan tiedostoonsa. Esimerkin kuvat tallennettiin tiedostoihin **`nuoli-v.gif`** (vasemman reunan nuoli), **`nuoli-k.gif`** (keskellä oleva nuoli) ja **`nuoli-o.gif`** (oikean reunan nuoli).

Seuraavaksi kuvat yhdistetään animaatioksi. **Gif Construction Set** -ohjelmassa aloitetaan uuden animaation tekeminen valitsemalla **`File`**-valikon toiminto **`New`**. Ohjelman kuvaruudulle tulee tyhjä tila, johon on valmiiksi täytetty animaation alustustiedot:



Alustustiedoissa määritellään eräitä tietoja animaatiosta. Kun näpäytät hiirellä kahdesti riviä, jossa on sana "**Header**", saat ruudullesi ikkunan, jossa voit muuttaa näitä tietoja:

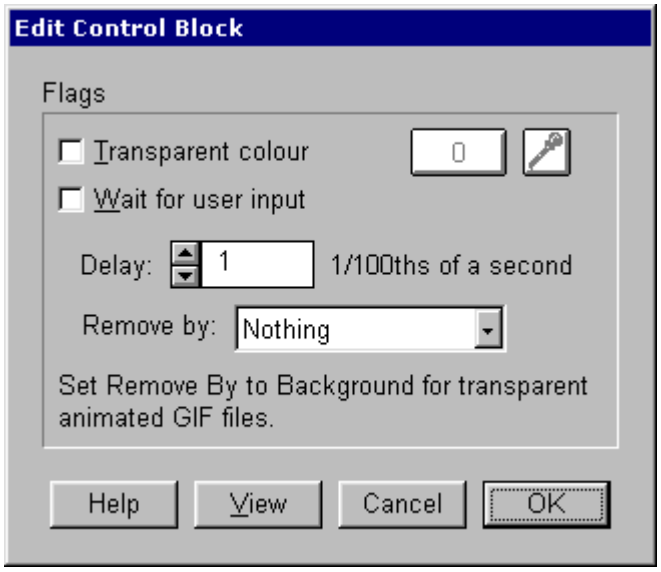


Kohtaan **Screen width** määritellään **Gif**-animaation leveys pikseleinä ja kohtaan **Screen depth** animaation korkeus niin ikään pikseleinä. Jokainen

esimerkin animaation nuolikuva oli kooltaan 100 x 50 pikseliä (animaatioiden jokainen kuva kannattaa tehdä samankokoiseksi), joten täytetään nämä arvot animaation kooksi. Paina lopuksi **OK**-painiketta.

Seuraavaksi lisätään määrittely, jolla animaatiosta tehdään silmukka. Se tapahtuu valitsemalla painike **Insert** ja sitten **Loop**. Määrittelyjen listalle tulee uusi rivi, jossa on teksti "**LOOP**". Kun näpäytät hiirellä kahdesti tätä riviä kuvaruudulle tulee ikkuna, jossa pyydetään määrittelemään animaation toistokertojen määrä. Oletusarvona on 1000, joka tarkoittaa käytännössä päättymätöntä animaatiota. Tähän voi määritellä vaikkapa numeron 10, jolloin ruudulle tuleva nuoli heilahtaa kymmenen kertaa ja pysähtyy sitten.

Seuraavaksi on aika lisätä varsinaiset animaation kuvat. Ennen jokaista kuvaa lisätään kontrollimäärittely. Siinä kerrotaan, miten kauan kuvaa pidetään ruudulla ennen kuin se korvautuu seuraavalla kuvalla ja miten uuden kuvan tulee korvata vanha kuva. Kontrollimäärittely lisätään valitsemalla ensin **Insert** ja sitten **Control**. Napsauta jälleen hiirellä kahdesti ruudulle tullutta riviä, jolloin voit tehdä lisämäärittelyitä:

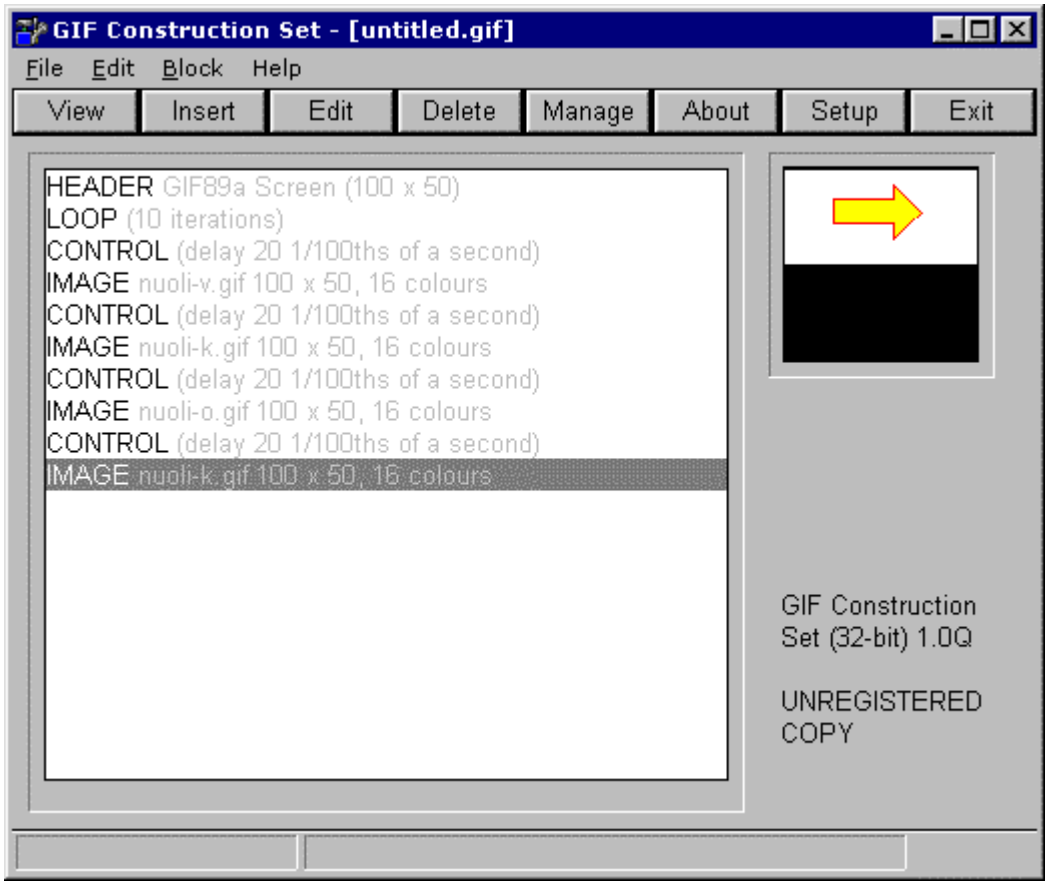


Kohdassa **Delay** määritellään sekunnin sadasosina, miten kauan kuva pysyy ruudulla. Esimerkin animaatiossa tähän laitetaan arvo 20, joka tekee liikkeestä sopivan pehmeän muttei liian nopean. Paina taas **OK**.

Itse kuva lisätään valitsemalla **Insert** ja **Image**. Ruudulle tulee näkyviin normaali tiedostonlatausikkuna, josta voit etsiä ensimmäisen piirtämistäsi kuvista. Aloitetaan vasemmalla olevasta nuolesta (**`nuoli-v.gif`**). Kun olet valinnut tiedoston, **Gif Construction Set** pyytää sinua määrittelemään väripaletin, jota käytetään animaatiossa:

Kaikki nuolikuvat piirrettiin siten, että ne käyttävät samaa 16 värin palettia. Sillä tavalla kuvista tuli pienikokoisia. Animaatio määritellään käyttämään tätä väripalettia valinnalla "**Use this image as the global palette**" (Käytä tämän kuvan väripalettia koko animaation väripalettina).

Tällä tavalla lisätään peräjälkeen kaikki animaatioon kuuluvat kuvat. Esimerkissä käytetään yhtä kuvaa (**`nuoli-k.gif`**) kahdesti, koska kuvat lisättiin järjestyksessä vasen-keski-oikea-keski:



Voit testata animaatiotasi napsauttamalla **View**-painiketta. Pääset takaisin animaation katselusta painamalla näppäimistöltäsi **Esc**-näppäintä. Voit muuttaa animaation nopeutta ja muita siihen liittyviä ominaisuuksia halutessasi editoimalla animaatiolistassa olevia **control**-kohtia.

Lopuksi animaatio tallennetaan **File**-valikon toiminnolla **Save As**. Tallenna tekemäsi animaatio vaikkapa nimellä **`animaatio.gif`**. Lopputuloksena saatu animaatio on **gif**-muodossa oleva tiedosto, jota käytetään aivan kuten tavallisia **gif**-kuvia: lisää animaatio sivusi sisään komennolla

```
<img src="animaatio.gif">
```

Animaation yksittäiset kuvat olivat kooltaan vain noin 260 tavua. Itse animaation kooksi tuli 1,3 kilotavua.

INTERAKTIIVISET KUVAT

Interaktiivisten, "klikattavien" kuvien avulla voit tehdä grafiikkaa sisältävistä sivuistasi entistä helppokäyttöisempiä tai näyttävämpiä. Interaktiivi-

nen kuva näyttää tavalliselta kuvalta. Se eroaa kuitenkin tavallisesta kuvasta: samoin kuin linkkejä sisältävä sivu sisältää "kuumia sanoja", interaktiivinen sivu sisältää "kuumia kohtia". Kun näpäytät hiiren näppäintä kuvassa olevan kuuman kohdan päällä, saat näytöllesi uuden WWW-sivun — samaan tapaan kuin tekstilinkeissäkin.

Interaktiivinen kuva tarjoaa sinulle uusia mahdollisuuksia: voit tehdä sivullesi esimerkiksi interaktiivisen kartan, jolloin sivun lukija saa kartan kohtaa näpäyttämällä näkyviin suurennuksen valitusta kartan kohdasta. Voit tehdä myös sivuillesi navigointipalkkeja tai muita käyttöliittymiä.

Mahdollisuudet ovat rajattomat — mielikuvitusta käyttämällä interaktiivisella kuvalla saa aikaan melkein mitä vain.

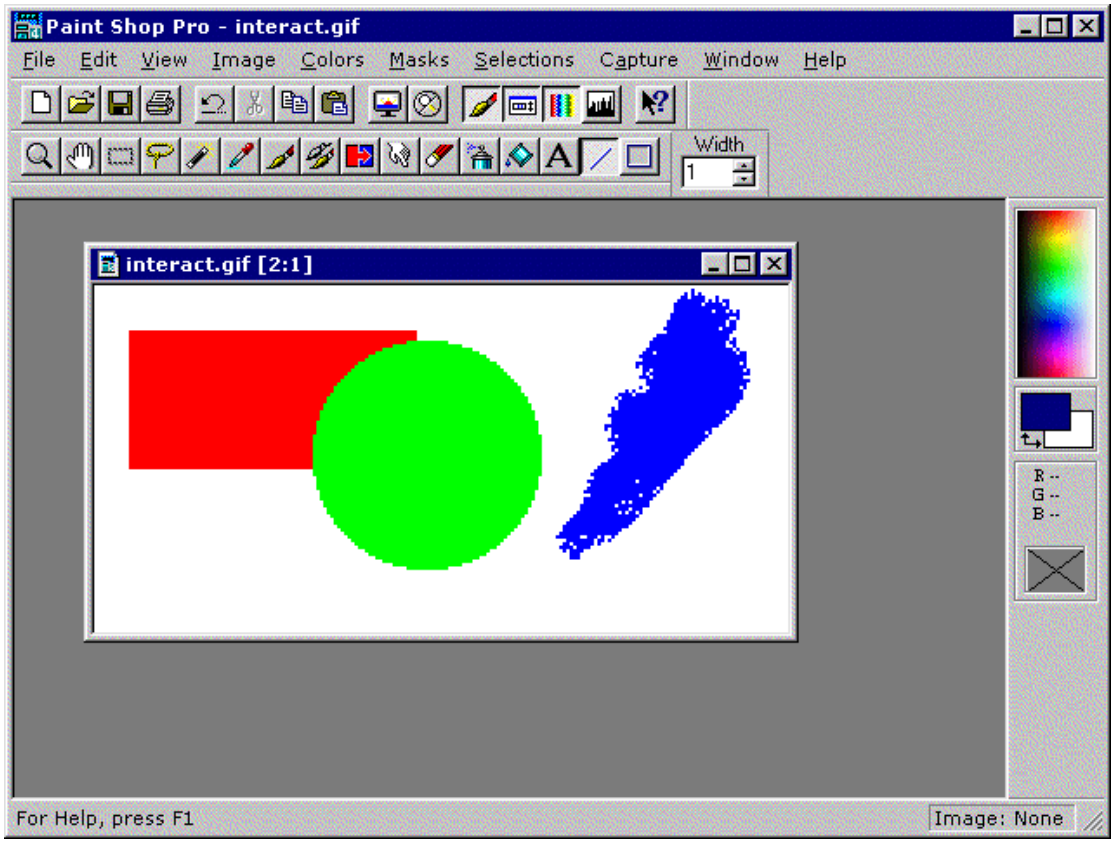
Jos innostut interaktiivisista kuvista, pysähdy kuitenkin hetkeksi ajattelemaan tekstipohjaisten selainten käyttäjiä. Heidän ohjelmillaan ei pysty katselemaan WWW-sivujen kuvia, jolloin myös interaktiiviset kuvat jäävät heiltä näkemättä. Myöskään ne käyttäjät, jotka ovat kytkeneet kuvien latauksen pois päältä, eivät näe interaktiivisia kuvia. Hyvässä WWW-sivussa onkin aina myös tekstipohjainen vaihtoehto.

Tekstivaihtoehdon toteuttamiseen on monta tapaa. Yleisin tapa on sijoittaa interaktiivisen kuvan alle kaikki kuvasta löytyvät linkit myös tavallisena tekstinä. Tällä tavalla ne, jotka eivät pysty näkemään kuvaa pääsevät kuitenkin käsiksi tarjolla olevaan tietoon. Toinen vaihtoehto on sijoittaa kuvan alle linkki toiselle sivulle, jossa kuvan sisältämät linkit on esitetty tekstimuodossa. Tämän linkin nimi voi olla vaikka "Sama sivu tekstimuodossa".

Tekstivaihtoehdosta on hyötyä ei-graafisten selaimien käyttäjien lisäksi myös niille, jotka eivät mahdollisesti keksineet näpäytellä hiiren näppäintä kuvan päällä. Jos kaikki interaktiivisen kuvan kuumat kohdat eivät ole selkeästi merkittyjä, tekstivaihtoehto auttaa myös löytämään kaikki mahdolliset tavat edetä sivulla eteenpäin.

KUVAKARTAN TEKEMINEN

Kun alat tehdä interaktiivista kuvaa, ensimmäiseksi sinun täytyy hankkia sopiva kuva. Kuvan täytyy olla **gif**- tai **jpeg**-muodossa, aivan kuten muidenkin sivuillasi käytettyjen kuvien. Tässä käytetään esimerkkinä kuvaa, joka sisältää punaisen nelikulmion, vihreän ympyrän ja sinisen epäsäännöllisen kuvion. Tallennetaan kuva HTML-tiedostojen kanssa samaan hakemistoon nimellä **`interact.gif`**.



Seuraavaksi vuorossa on kuumien kohtien suunnittelu. Ne voivat olla nelikulmaisia, pyöreitä tai monikulmion muotoisia alueita.

Kuumien kohtien etsimiseen tarvitset ohjelman, jolla voit laskea kuvan koon pikseleinä sekä etsiä kuvan eri kohtien koordinaatit. Tällaisia ohjelmia ovat esimerkiksi monet piirustus- ja kuvankäsittelyohjelmat. Esimerkiksi **Paint Shop Pro** osaa näyttää kuvasta koordinaatit.

Tarkoituksena olisi tehdä interaktiivisesta kuvasta sellainen, että jokaista värillistä kohtaa napsauttamalla lukija pääsee eri sivulle. Interaktiivisten kohtien kohteet ovat samoja kuin tavallisissakin linkeissä: ne voivat olla joko itse tehtyjä HTML-tiedostoja tai URL-osoitteita, jotka osoittavat muualla Internetissä sijaitseville sivuille.

Ensimmäisenä suunnitellaan sivut, joihin lukija pääsee napsauttamalla jotain kuvan kohtaa hiirellä ja kirjoitetaan ne tiedostoihin tai valitaan verkosta osoitteet, joihin hiiren napsautukset vievät lukijan. Esimerkkikuvaan laitetaan seuraavat määrittelyt:

- Punaisen nelikulmion napsautus vie **C|Netin** kotisivulle osoitteeseen **`http://www.cnet.com/`**

- Vihreän ympyrän napsautus vie itse tehtyyn tiedostoon nimeltään **kotisivu.htm**.

- Sinisen kuvion napsautus vie **Yahoo**-palvelin sivuille, osoitteeseen **http://www.yahoo.com**

Kun kuumat kohdat on valittu, niistä tehdään HTML-sivun sisään kuvakartta. Se aloitetaan komennolla **<map>**. Komennon sisään laitetaan lisämäärittelyiksi kyseisen karttatiedoston nimi attribuutilla **name**. Nimen avulla karttatiedostoon viitataan myöhemmin varsinaisista kuvista. Esimerkiksi:

```
<map name="kuviot">
```

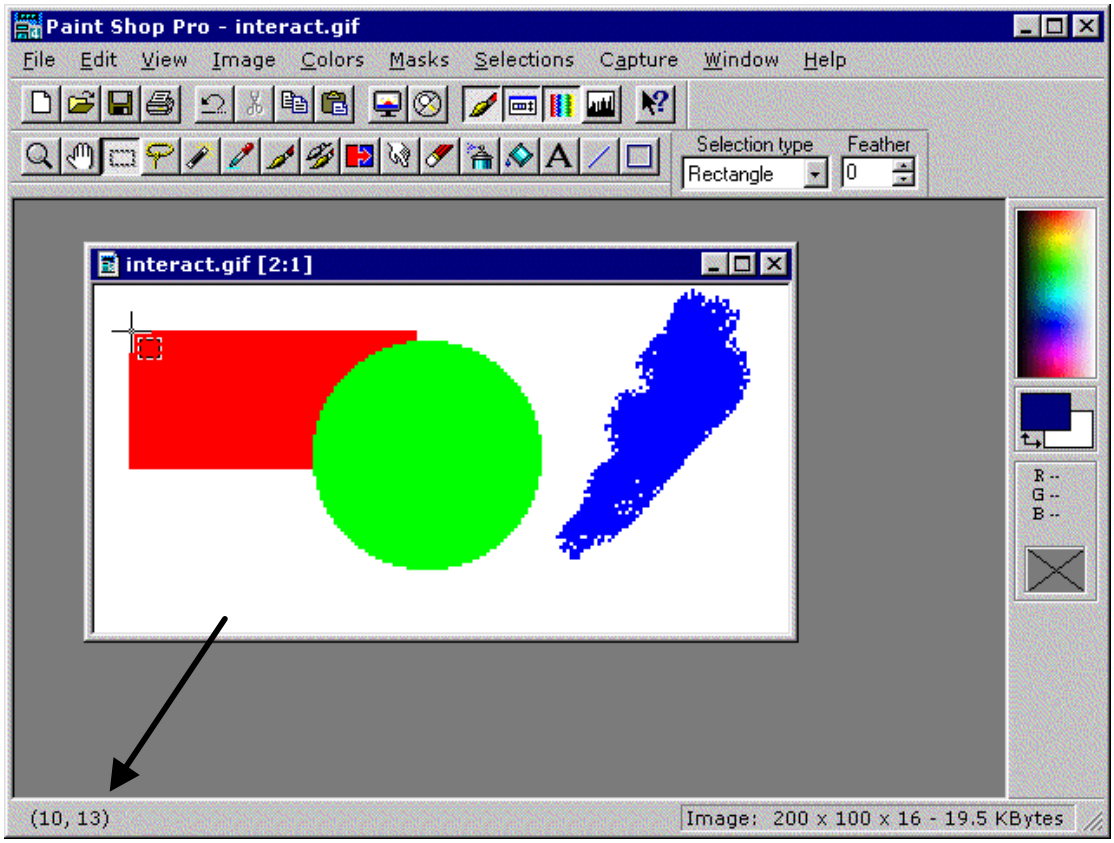
Kuvakartta lopetetaan komentoon **</map>**. Aloitus- ja lopetuskomennon väliin sijoitetaan varsinaiset kuumien kohtien määrittelyt. Jokainen määrittely merkitään komennolla **<area>**. Komento saa attribuutteja, jotka kertovat halutun alueen muodon (**shape**) ja sen koon (**coords**). Lisäksi **<area>**-komentoon määritellään attribuutilla **href** sivu, joka on tarkoitus ladata käyttäjän näpäytettyä kyseistä kohtaa.

Nelikulmion muotoisen alueen attribuutti on **rect**. Nelikulmiosta ilmoitetaan sen vasemman ylänurkan ja oikean alanurkan koordinaatit. Esimerkkikuvassa neliön vasen ylänurkka on kohdassa 10,13 ja oikea alakulma kohdassa 93,53. Nelikulmiota napsauttamalla siirryttiin **C|Netin** kotisivulle. Kuvakarttaan sijoitetaan siis seuraavanlainen komento.

```
<area shape="rect" coords="13,13,95,53"
href="http://www.cnet.com/">
```

Huomaa, kuinka koordinaatit kirjoitetaan peräkkäin pilkuilla erotettuina. Attribuutti **href** kertoo osoitteen, joka ladataan käyttäjän näpäytettyä hiirellä kyseistä kohtaa.

Koordinaattien etsiminen **Paint Shop Prossa** sujuu näppärästi. Kytke ensin päälle jokin piirtotyökalu ja vie sen jälkeen työkalun kärki haluamasi kohdan päälle. Näet kyseisen paikan koordinaatit ikkunan vasemmassa alanurkassa:



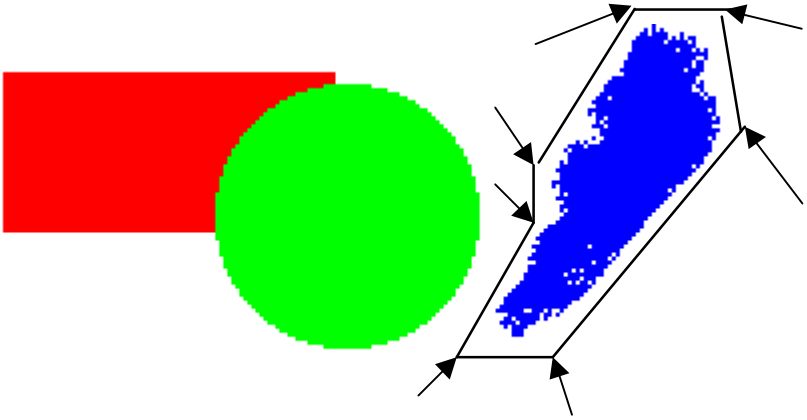
Ympyränmuotoinen alue määritellään antamalla alueen muodoksi **`circle`**. Ympyrän koko määritellään antamalla sen keskipisteen koordinaatit sekä sen säde. Esimerkkinä **`<area>`**-komento esimerkin ympyrälle näyttää seuraavalta:

```
<area shape="circle" coords="95,48,33"
href="kotisivu.htm">
```

Ympyrän keskipiste oli kohdassa 95,48; sen säde oli 33 ja kohdan napsautus tuo näkyviin HTML-tiedoston **`kotisivu.htm`**.

Kun haluat tehdä erikoisemman muotoisia kuumia kohtia, voit määritellä niiden muodoksi **`poly`**. Tällöin voit määritellä itse alueen muodon antamalla haluamaasi aluetta reunustavan monikulmion koordinaatit yksi kerrallaan.

Kuvittele haluamasi alueen ympärille suorista viivoista piirretty "aita". Kun valitset jokaisen aidan kulman koordinaatit (kuvassa merkitty nuolilla), saat **`poly`**-attribuuttiin sopivat koordinaattilistan.



Esimerkki monikulmion tekevästä komennosta:

```
<area shape="poly" coords="129,81, 130,62, 143,45,
144,26, 163,1 184,1, 195,24, 144,83"
href="http://www.yahoo.com/">
```

Kun käyttäjä näpäyttää hiirellään kuvan taustaa, eli paikkaa, johon ei ole määritelty kuumaa kohtaa, selain ei reagoi painallukseen mitenkään. Jos haluat, että taustaa näpäyttämällä ladataan jokin tietty sivu, voit määritellä sen antamalla alueen muodoksi määrittelyn **default**. Esimerkki:

```
<area shape="default" href="http://www.news.com/">
```

Jos haluat määritellä alueita, joista näpäyttämällä ei tapahdu mitään, voit määritellä ne antamalla komennossa osoitteen sijasta attribuutin **nohref**. Tällä tavalla voit toteuttaa erikoisemman muotoisia kuumia kohtia. Voit tehdä esimerkiksi munkkirinkilän muotoisen kuuman kohdan tekemällä ensin suuren ympyrän ja sijoittamalla sen sisään pienemmän ympyrän, jota näpäyttämällä ei tapahdu mitään.

Karttatiedosto liitetään kuvaan lisäämällä kuvakomentoon attribuutti **usemap**. Sen sisällöksi annetaan linkki karttamäärittelyjen **<map>**-komennossa annettuun nimeen. Valmis kuvakartta ja sitä käyttävä kuva näyttävät siis seuraavalta:

```
<map name="kuviot">
<area shape="circle" coords="95,48,33"
href="kotisivu.htm">
<area shape="rect" coords="13,13,95,53"
href="http://www.cnet.com/">
```

```
<area shape="poly" coords="129,81, 130,62, 143,45,
144,26, 163,1 184,1, 195,24, 144,83"
href="http://www.yahoo.com/">
</map>

<h1>Interaktiivinen kuva!</h1>
<p>Kokeile napsautella kuvan eri kohtia.
<p><img src="interact.gif" usemap="#kuviot">
```

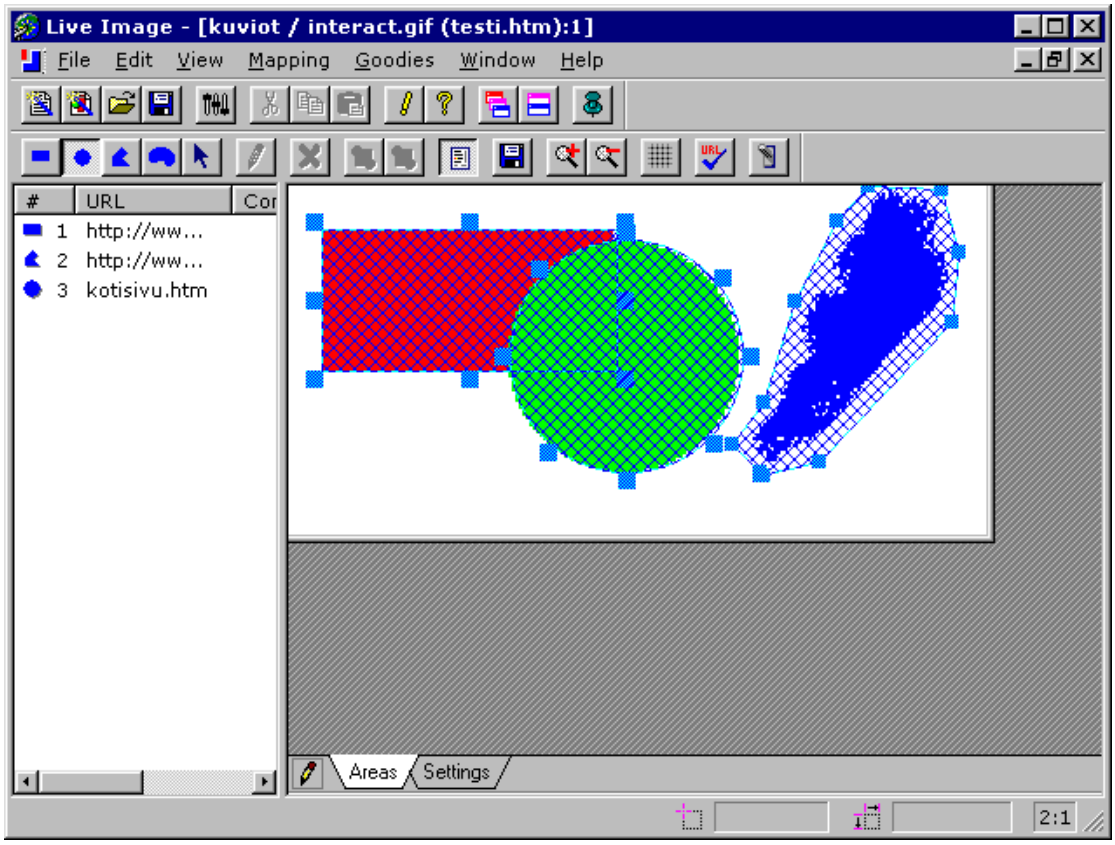
Huomaa, että rivien järjestys on tärkeä silloin, kun sinulla on päällekkäin meneviä kuumia kohtia. Esimerkissä vihreä ympyrä peittää vähän punaisesta nelikulmiosta. Alueet luetellaan järjestyksessä päällimmäisestä alimmaiseen, joten nelikulmion päällä oleva ympyrä täytyy mainita määrittelyissä ennen nelikulmiota.

Interaktiivisen kuvan ympärille tulee näkyviin sininen kehys aivan kuten silloin, kun teet kuvasta linkin. Voit poistaa kehyksen lisäämällä kuvakomentoon attribuutin **`border=0`**.

APUOHJELMAT INTERAKTIIVISTEN KUVIEN TEKEMISEEN

Interaktiivisten kuvien tekeminen saattaa käydä rasittavaksi, jos kuvassa on paljon ja monimutkaisia kuumia kohtia. Verkosta löytyvillä apuohjelmilla kuumien kohtien määrittely käy kuitenkin helposti hiiren avulla.

Yksi monista apuohjelmista on nimeltään **LiveImage**. Voit imuroida sen itsellesi sivulta **`http://www.mediatec.com`**



LiveImage-ohjelman avulla voit piirtää hiirellä kuvaasi kuumat kohdat ja määritellä osoitteet, joihin niiden napsautus vie. Ohjelma osaa lisäksi tehdä puolestasi kuvakartan ja sijoittaa sen HTML-tiedoston sisään.

Lisää interaktiivisten kuvien tekemiseen tarkoitettuja apuohjelmia löytyy monista Internetin ohjelma-arkistoista, esimerkiksi **Tucows:ista** osoitteesta

`http://www.eunet.fi/tucows/imap95.html` .

POSTIMERKKIKUVAT

Jos haluat tarjota samasta kuvasta sekä pienen, kaikille näytöille ja kaikille värimäärille sopivan version, että suuremman värikkään version niille käyttäjille, joiden laitteistokapasiteetti riittää tällaisten kuvien katselemiseen, voit tehdä nk. postimerkkikuvan.

Postimerkkikuvaksi kutsutaan pientä sivulle sijoitettua kuvaa, jota näpäyttämällä saadaan näkyviin sama kuva suurempana. Pieni kuva ei vie paljon tilaa sivulta ja sen lataaminen sujuu nopeasti. Ne, jotka haluavat tarkastella kuvaa tarkemmin, voivat ladata isomman kuvan ihailtavakseen.

Postimerkkikuvat toteutetaan tekemällä pienemmästä kuvasta linkki suurempaan kuvaan. Linkin "kuumaksi sanaksi" sijoitetaankin tekstin sijasta pienempi kuvista. Linkin osoittama tiedosto on kuva alkuperäisessä suuressa koossaan.

Voit myös ottaa suuremmasta kuvasta osan ja käyttää sitä pienennetyn kuvan sijasta. Voit ottaa vaikkapa suuresta kuvasta jonkin yksityiskohdan ja houkutella sen avulla lukijan katselemaan isompaa kuvaa.

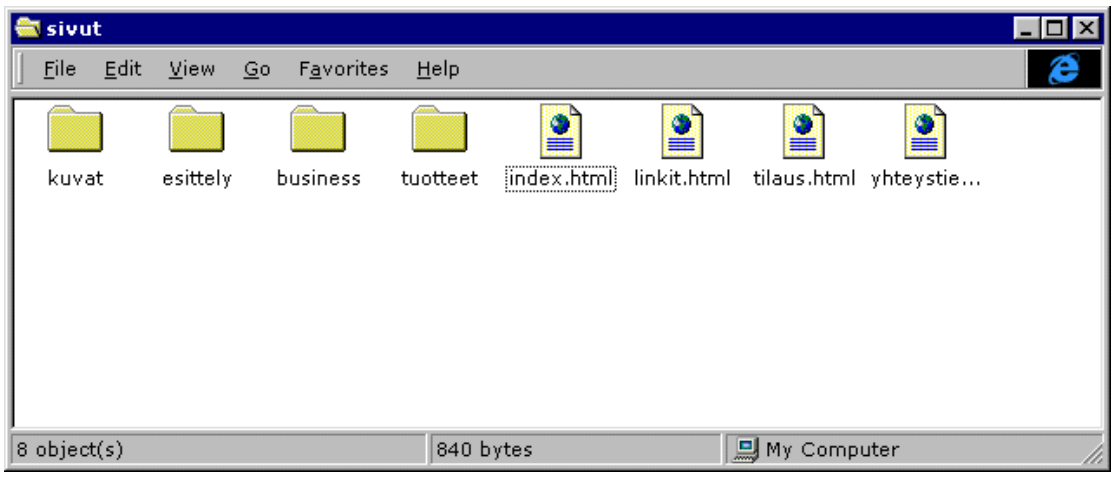
TIEDOSTOJEN KÄSITTELY

Kun WWW-sivusi kasvavat, niihin liittyy yhä enemmän ja enemmän tiedostoja. Jokainen sivu saattaa sisältää useita kuvia ja itse sivujakin voi olla paljon.

Monimutkaisille sivuille kannattaakin suunnitella hakemistorakenne, jonka avulla sivut pysyvät järjestyksessä. Tässä luvussa kerrotaan hakemistorakenteen suunnittelusta ja linkkien tekemisestä hakemistorakenteen sisällä.

HAKEMISTOT

Voit tehdä HTML-hakemistoosi alihakemistoja samalla tavalla kuin teet muitakin hakemistoja tietokoneessasi. Kun teet laajoja sivukokonaisuuksia, jokaiselle asiakokonaisuudelle kannattaakin tehdä oma alihakemistonsa.

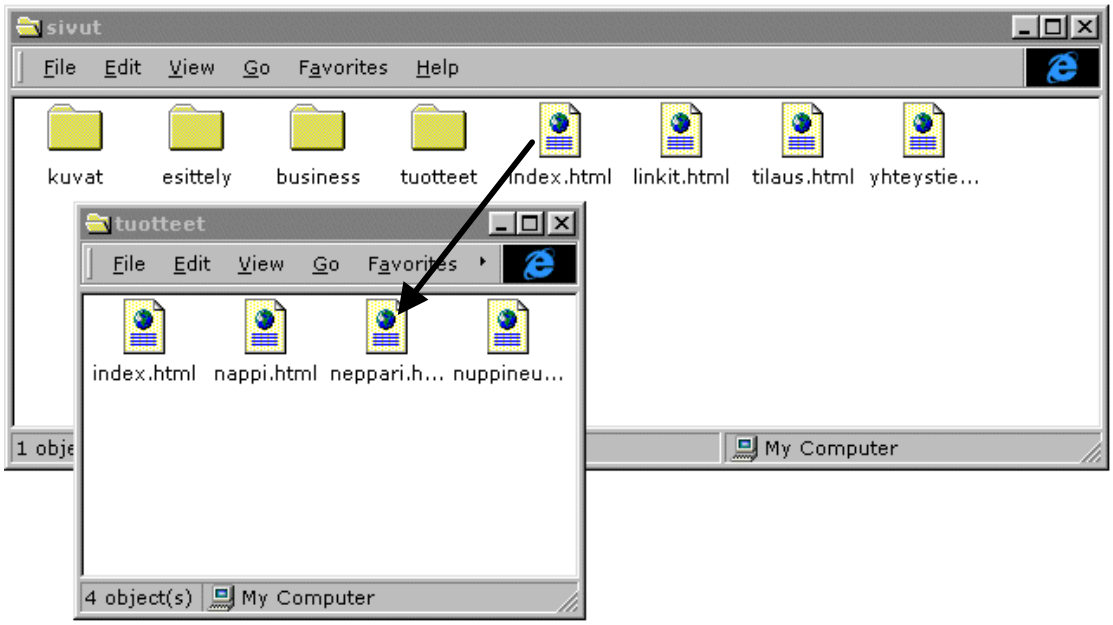


Yllä olevassa kuvassa on esimerkkinä kuvitteellisen yrityksen kotisivut. Päähakemistossa (kuvassa) on vain neljä HTML-tiedostoa, joissa on yrityksen varsinainen kotisivu sekä sivut linkkilistalle, tilauslomakkeelle ja yhteystiedoille.

Kolmessa alihakemistossa on erikseen HTML-sivuja yrityksen esittelylle, business-sivuille ja tuotteiden esittelyille. Lisäksi kaikki WWW-sivujen kuvat on siirretty yhteen hakemistoon.

Voit viitata linkeissä ja kuvakomennoissa hakemistosta toiseen liittämällä tiedoston nimeen mukaan polun. Oletetaan, että yrityksen tuotevalikoimaan kuuluu nappeja, neppareita ja nuppineuloja, joista jokaiselle on tehty oma sivunsa tuotteet-alihakemistoon.

Yrityksen etusivulta (**index.html**) halutaan tehdä linkki neppareita esittelevälle sivulle (**neppari.html** alihakemistossa **tuotteet**):



Tällainen linkki tehdään liittämällä tiedoston nimeen hakemiston nimi:

```
<p>Suosituin tuotteemme tällä hetkellä on useita
palkintojakin voittanut
<a href="tuotteet/neppari.html">Neppari</a>.
```

Jos haluat tehdä linkin alihakemistosta ylempään hakemistoon, esimerkiksi tiedostosta **neppari.html** takaisin etusivun tiedostoon **index.html**, voit tehdä sen lisäämällä tiedoston nimeen edellisen hakemiston tunnuksen. Edellisen hakemiston tunnus on kaksi pistettä (**..**):

```
<p><a href="../index.html">Takaisin etusivulle</a>.
```

Koska WWW-sivut sisältävät usein paljon kuvia, on hyvä ajatus sijoittaa kaikki kuvat omaan hakemistoonsa. Tällä tavalla HTML-tiedostoja sisältävät sivut pysyvät selkeämpinä, kun kuvat on kerätty omaan paikkaansa.

Kun haluat käyttää alihakemistossa olevaa kuvaa, laita kuvakomentoon hakemiston nimi samalla tavalla kuin linkkikomentoonkin:

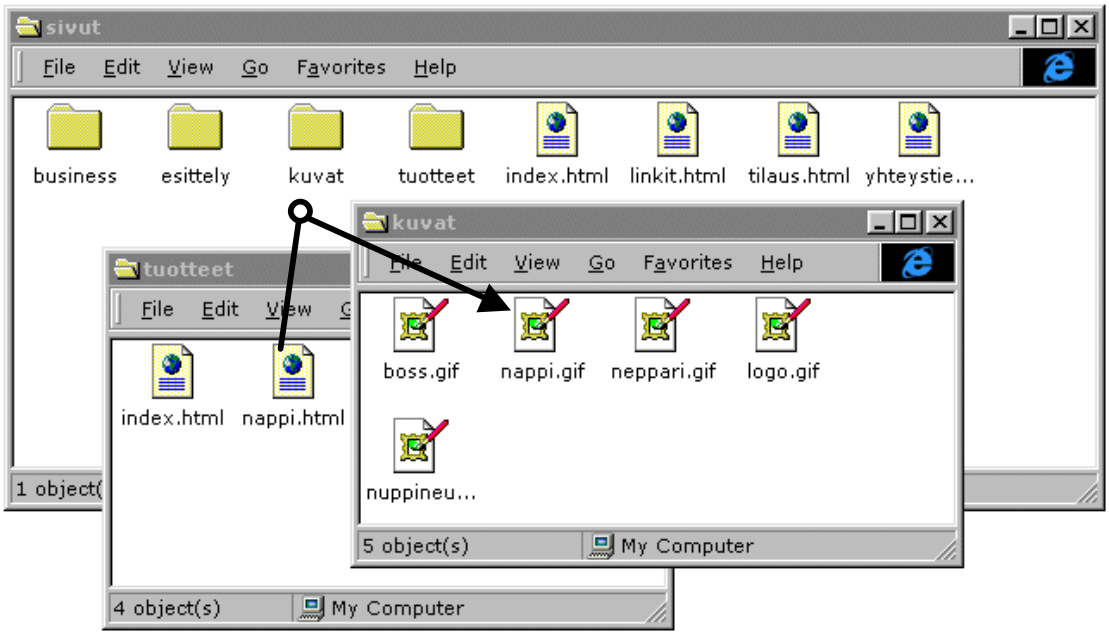
```
<img src="kuvat/logo.gif">
```

Jos haluat käyttää alihakemistossa sijaitsevalta sivulta rinnakkaisessa hakemistossa olevaa tiedostoa, esimerkiksi kuvaa, sinun täytyy määritellä hakupolku kulkemaan ylemmän hakemiston kautta.

Esimerkiksi, jos haluat käyttää tiedostossa **tuotteet/nappi.html** kuvaa **kuvat/nappi.gif**, kirjoita kuvakomennon tiedoston nimeen aluksi polku "**..**", joka siirtyy hakemistosta **tuotteet** takaisin päähakemistoon. Kirjoita tiedostonnimeen tämän jälkeen hakemiston **kuvat** nimi ja lopuksi tiedostonnimi **nappi.gif**:

```
<img src="../kuvat/nappi.gif">
```

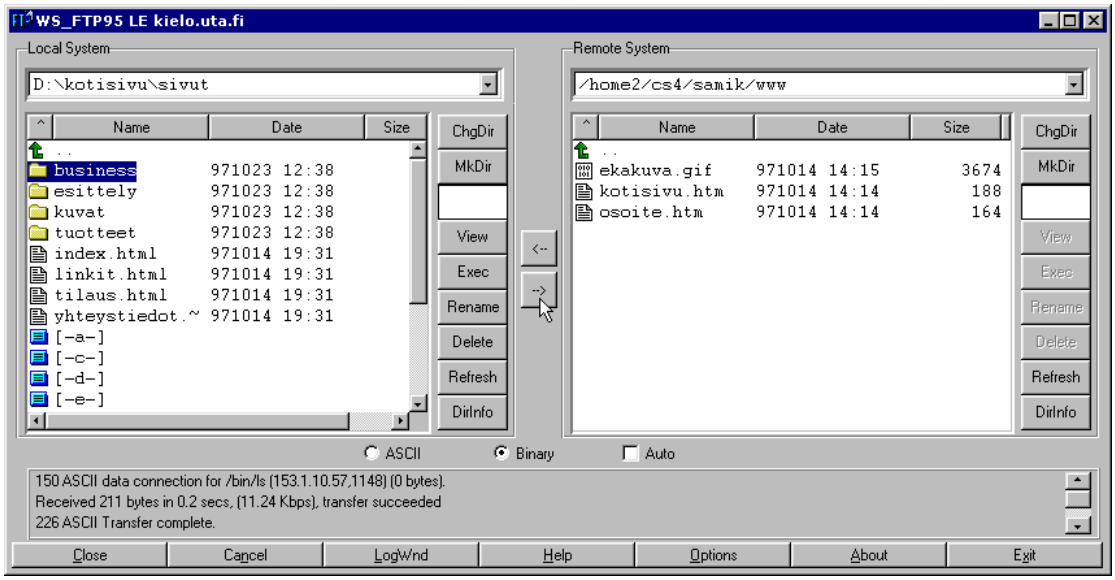
Kuvassa näet, miten viittaus siirtyy kuvat-hakemistoon edellisen hakemiston kautta.



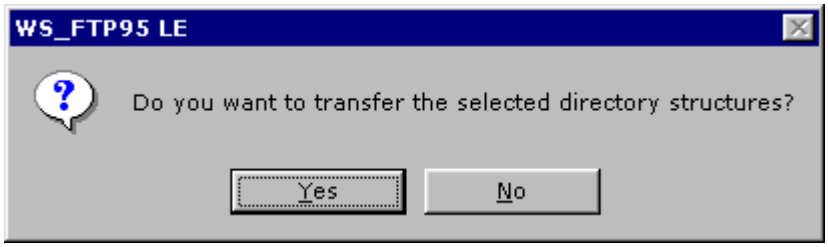
HAKEMISTOT JA URL-OSOITTEET

Kun siirrät hakemistorakennettasi palvelinkoneelle, siirrä hakemiston ja niiden sisältö sellaisenaan. Tällä tavalla linkkisi säilyvät toimivina.

Voit siirtää kokonaisia hakemistoja **WS-FTP**:llä napsauttamalla ensin tiedostolistauksesta hakemiston ikonia ja sen jälkeen siirron aloittavaa nuolta:



WS-FTP kysyy sinulta haluatko siirtää hakemiston ja sen sisällön palvelinkoneelle. Vastaa tähän **Yes**:



Sivujesi hakemistorakenne siirtyy suoraan myös sivujesi URL-osoitteisiin. Jos kotisivusi osoite on esimerkiksi **`http://www.oper.fi/~tunnus/`** ja kotisivuillasi on esimerkiksi alihakemisto nimeltään "**`tuotteet`**" ja sen sisällä tiedosto nimeltään **`nappi.html`**, sen osoitteeksi tulee:

```
http://www.oper.fi/~tunnus/tuotteet/nappi.html
```

Kun sivuillasi on linkkejä, joissa on annettu alihakemistojen nimiä tai edellisen hakemiston tunnus (..), selain osaa muokata osoitteet automaattisesti oikein. Esimerkiksi jos sivulla **`http://www.oper.fi/~tunnus/tilaus.html`** on linkki

```
<a href="tuotteet/nappi.html">
```

selain osaa automaattisesti siirtää käyttäjän sivulle **http://www.oper.fi/~tunnus/tuotteet/nappi.html** linkkiä napsautettaessa.

Voit tehdä hakemistorakenteestasi niin syvän kuin haluat. Sinulla voi olla esimerkiksi hakemisto "business", joka sisältää alihakemistot "kotimaa" ja "ulkomaat". Hakemiston "ulkomaat" sisällä voi olla lisäksi alihakemistoja, esimerkiksi "eurooppa" ja "aasia":

Tällöin eri hakemistoihin osoittavat URL-osoitteet voisivat olla alla olevan näköisiä. Tutki osoitteita ja vertaa niitä yllä näkyvään kuvaan, niin huomaat miten osoitteet muodostuvat hakemistojen nimien perusteella.

http://www.oper.fi/~tunnus/business/yritysidea.html

Tiedosto **yritysidea.html** kansiossa **business**.

http://www.oper.fi/~tunnus/business/kotimaa/lappi.html

Tiedosto **lappi.html** kansiossa **kotimaa**.

http://www.oper.fi/~tunnus/business/ulkomaat/
aasia/japani.html

Tiedosto **japani.html** kansiossa **aasia** (yllä näkyvä osoite on jaettu kahdelle riville, jotta se mahtuu kirjan sivulle - selaimessa osoitteet kirjoitetaan aina yhdelle riville).

Kun teet linkkejä hakemistoista toisiin, ilmoita alaspäin rakennetta mennessäsi kansioiden nimet. Ylöspäin rakennetta mennessäsi voit käyttää edellisen hakemiston tunnusta, joka on kaksi pistettä (..). Esimerkkejä:

Linkki päähakemistosta sivulle **aasia/japani.html** tehdään näin:

```
<a href="business/ulkomaat/
aasia/japani.html">
```

Linkki tiedostosta **eurooppa/ruotsi.html** pääsivulle tiedostoon **index.html** tehdään näin:

```
<a href="../../../index.html">
```

Huomaa, kuinka jokainen kahden pisteen muodostama hakemiston nimi hyppää yhden pykälän ylöspäin rakenteessa: ensimmäinen vie hakemistosta eurooppa hakemistoon ulkomaat, seuraava hakemistosta ulkomaat hakemistoon business ja viimeinen hakemistosta business päähakemistoon.

Tarvittaessa voit myös yhdistellä tuplapisteitä ja hakemistojen nimiä, esimerkiksi silloin jos haluat tehdä linkin tiedostosta **eurooppa/suomi.html** tiedostoon **tuotteet/nappi.html**:

```
<a href=
"../../../tuotteet/nappi.html">
```

Huomaa, kuinka uuteen alihakemistoon kuljetaan päähakemiston kautta.

Kun teet monimutkaisia hakemistorakenteita, huomaat varmasti pian että sivujesi URL-osoitteet muuttuvat pitkiksi. Tällaisia osoitteita on hankala muistaa ja hankala kirjoittaa.

Jos aiot julkistaa maailmalle osoitteita, jotka vievät suoraan johonkin sivuillasi olevaan alihakemistoon, kiinnitä silloin huomiota hakemistorakenteeseesi. Yritä sijoittaa sivut, joita tulet mainostamaan, mahdollisimman lähelle sivujesi alkupistettä niin, että sivujesi osoitteet eivät muutu liian monimutkaisiksi.

Voit lyhentää osoitteita myös tekemällä jokaiselle hakemistolle oman etusivutiedoston. Kun annat HTML-tiedostolle nimeksi **index.html** (tai **index.htm**), voit jättää sen kokonaan pois osoitteista. Esimerkiksi nämä kaksi osoitetta tarkoittavat samaa tiedostoa:

```
http://www.oper.fi/~tunnus/tuotteet/
```

```
http://www.oper.fi/~tunnus/tuotteet/index.html
```

Joidenkin Internet-operaattorien palvelimissa nimi **index.html** ei toimi, vaan hakemiston etusivusta käytetään jotain muuta nimitystä (esimerkiksi **home.html**). Huomaat tämän jos saat virheilmoituksen yrittäessäsi käyttää ylemmäntyyppistä osoitetta. Saat lisätietoja käytettävästä nimestä oman operaattorisi ohjeista tai heidän asiakaspalvelustaan.

URL-osoitteet voivat olla myös palvelimen juuren suhteen relatiivisia. Tämä tarkoittaa sitä, että voit jättää osoitteesta pois alkuosan (**http://koneennimi**) ja käyttää tätä osoitetta linkkinä. Tällöin osoitteen täytyy kuitenkin alkaa kauttaviivalla. Yllä olevan esimerkin kanssa samaa tiedostoa tarkoittava osoite on myös

```
/~tunnus/tuotteet/index.html
```

Tätä osoitetta voit kuitenkin käyttää vain linkeissä ja kuvakomennoissa - älä siis julkista sivujesi osoitetta tällaisena.

TIEDOSTONNIMEN TARKENTIMET

Tiedostojen tarkentimet kertovat ohjelmille, minkä tyyppinen tiedosto on kyseessä. Tämän vuoksi on tärkeää, että käytät aina oikeaa tarkenninta tehdessäsi WWW-sivuja.

Jos annat tiedostollesi vääränlaisen tarkentimen, sivut eivät välttämättä toimikaan haluamallasi tavalla. HTML-sivut saattavatkin näkyä HTML-koodina tai kuvat tulevat näytölle rikkinäisinä.

HTML-tiedostojen pääte on **.html**. Koska vanhemmissa Windows-ympäristöissä ei voi käyttää kuin kolmikirjaimisia tarkentimia, myös **.htm** tarkoittaa HTML-tiedostoa.

Kuvatiedostojen päätteet riippuvat kuvaformaatista: **gif**-kuvien tarkennin on **.gif** ja **jpeg**-kuvien **.jpg**. Myös tarkennin **.jpeg** tarkoittaa **jpeg**-kuvaa.

Jos haluat sijoittaa sivuillesi puhtaita tekstitiedostoja, voit antaa niille tarkentimeksi **.txt**. Tällaiset tiedostot tulevat lukijan näytölle sellaisenaan.

Palvelimesi saattaa tunnistaa myös muita tiedostonnimen tarkentimia ja ohjata selaimen käsittelemään niitä halutulla tavalla. Jos sijoitat sivuillesi linkin ajettavaan ohjelmaan (tarkennin **.exe**), palvelin ja selain osaavat keskenään määritellä tyypiksi ohjelman. Tällöin käyttäjän selain osaa automaattisesti linkkiä seurattaessa kysyä, haluaako käyttäjä ajaa ohjelman vai tallentaa sen kovalevylle.

LOMAKKEET

Lomakkeiden avulla voit tehdä sivuillesi interaktiivisia osia. Niillä voit toteuttaa mitä erilaisimpia toimintoja aina henkilötietojen kyselystä vaikkapa erilaisiin tilauskaavakkeisiin.

Lomakkeeseen kuuluu kaksi osaa — itse lomake, joka tehdään sivulle HTML-komennoilla sekä lomakkeen tiedot käsittelevä ohjelma.

Kun sivun lukija täyttää lomakkeella annetut tiedot ja lähettää ne eteenpäin, tiedot täytyy käsitellä jollain tavalla. Tähän tarkoitukseen tarvitaan siis ohjelma, joka käsittelee lomakkeen tiedot. Riippuen siitä mitä tarkoitusta varten teet lomakkeita, ohjelma voi olla itse tehty tai jokin verkossa valmiina tarjolla oleva ohjelma. Voit myös käyttää menetelmää, jossa lomakkeen sisältö lähetetään vastaanottajalle sähköpostina. Alla olevissa esimerkeissä käytetään tätä menetelmää.

LOMAKEALUE

Form-komentojen välissä määritellään alue, joka sisältää tekstikenttiä, valikoita, painikkeita ja muita tekstin ulkoasuun vaikuttavia HTML-komentoja. Yhdellä WWW-sivulla voi olla useita lomakkeita: jokainen erillinen lomake ympäröidään **form**-komennoilla.

Form-komennolla määritellään **action**-attribuutilla osoite, jonne lomakkeelle syötetyt tiedot lähetetään käsiteltäviksi. Osoite voi olla joko ohjelmaan osoittava URL-osoite tai osoite, jonka avulla lomakkeelle syötetyt tiedot lähetetään eteenpäin sähköpostilla. Lisäksi komentoon liitetään attribuutti **method**, joka kertoo missä muodossa tiedot lähetetään.

Seuraavassa on esimerkki **form**-komentojen käytöstä:

```
<form action="mailto:webmaster@osoite.fi"
method=POST">
</form>
```

Yllä oleva esimerkki on yksinkertaisin mahdollinen lomake — se ei sisällä yhtään tekstiä eikä siinä ole valikoita, painikkeita tai tekstikenttiä. Esimer-

kin lomakkeessa on määritelty tietojen lähetystavaksi sähköposti ja vastaanottajan sähköpostiosoitteeksi **webmaster@osoite.fi**. Sähköpostilähetystä käytettäessä lomakkeen lähetysmuodoksi (**method**) asetetaan POST.

Näiden **form**-komentojen väliin sijoitetaan lomakkeen kentät ja painikkeet.

ENSIMMÄINEN LOMAKKEESI

Lomakkeeseen voi sijoittaa kahdenlaisia tekstikenttiä. Lyhyempiä tietoja varten on olemassa rivin mittaisia tekstilaatikoita sekä useamman rivin teksteille kokonaisia tekstialueita, jonne voi kirjoittaa pitempiä tekstejä.

Input-komento tuo lukijalle näkyviin yhden rivin korkuisen tekstilaatikon, johon hän voi kirjoittaa haluamaansa tekstiä. Voit tehdä tekstilaatikon, joka kysyy lukijan nimeä seuraavasti:

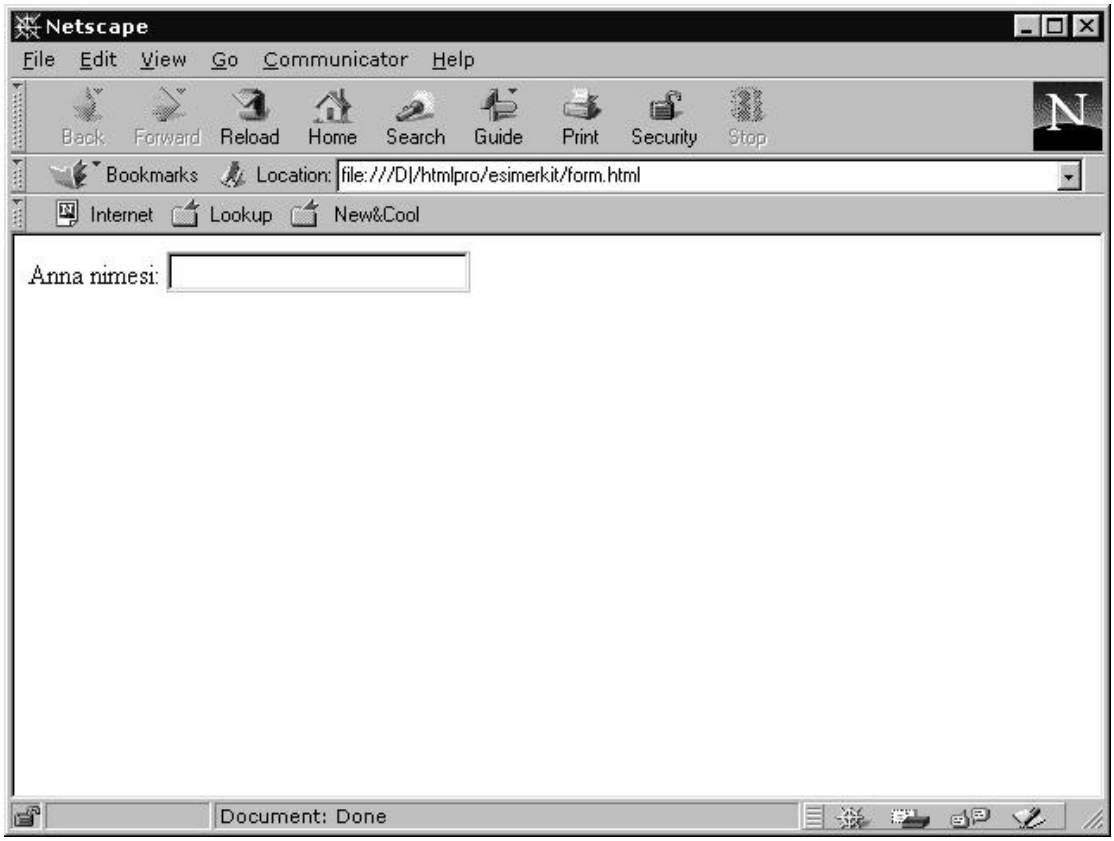
```
<form action="mailto:webmaster@osoite.fi"
method="POST">
<p>Anna nimesi:
<input type=text name="Nimi">
</form>
```

Jokaisella tekstilaatikolla on oma nimensä, joka määritellään attribuutilla **name**. Tämä nimi tulee näkyviin lomakkeen tietoja käsittelevässä ohjelmassa. Jos lomakkeen tiedot lähetetään eteenpäin vaikkapa sähköpostina, laatikon nimi tulee näkyviin postin lukijalle.

Tekstilaatikon tyyppi määritellään attribuutilla **type**. Tavallisimpia ovat laatikot, joihin voi kirjoittaa tavallista tekstiä. Niiden tyyppi on **text**, kuten yllä näkyvässä esimerkissä.

Itse **input**-komento tuo ruudulle vain pelkän laatikon. Tämän vuoksi sinun täytyy kirjoittaa sivulle tavallisena tekstinä se teksti, jonka haluat näkyvän laatikon vieressä (kuten esimerkin "Anna nimesi").

Tekstilaatikko tulee näkyviin selaimessa seuraavan näköisenä:



Lomake tarvitsee myös painikkeen, josta siihen annetut tiedot voidaan lähettää. Tämä tapahtuu lisäämällä sivulle **input**-komento, jonka tyyppi on **submit**.

```
<form action="mailto:webmaster@osoite.fi"
method=POST">
<p>Anna nimesi:
<input type=text name="Nimi">
<p>Lähetä lomakkeen tiedot painamalla nappia:
<input type=submit>
</form>
```

Tässä on ensimmäinen lomakkeesi! Voit kokeilla sen toimintaa itse. Muista kuitenkin vaihtaa esimerkin sähköpostiosoitteen tilalle oma sähköpostiosoitteesi. Sinun täytyy myös määritellä selaimessasi sähköpostin lähetykseen liittyvät asetukset (lisätietoa näistä asetuksista saat Internet-operaattoriltasi).

Saat hetkisen kuluttua postilaatikkoosi sähköpostia, jossa on täyttämäsi lomakkeen sisältö. Viestissä näkyy ensimmäisenä tekstilaatikon nimi, sen jälkeen yhtäsuuruusmerkki ja sitten lomakkeeseen syöttämäsi teksti:

```
Nimi=Sami
```

Jos kirjoitit tekstikenttään välilyöntejä tai skandinaavisia merkkejä, huomaat että saamasi sähköposti näyttää hieman erikoiselta:

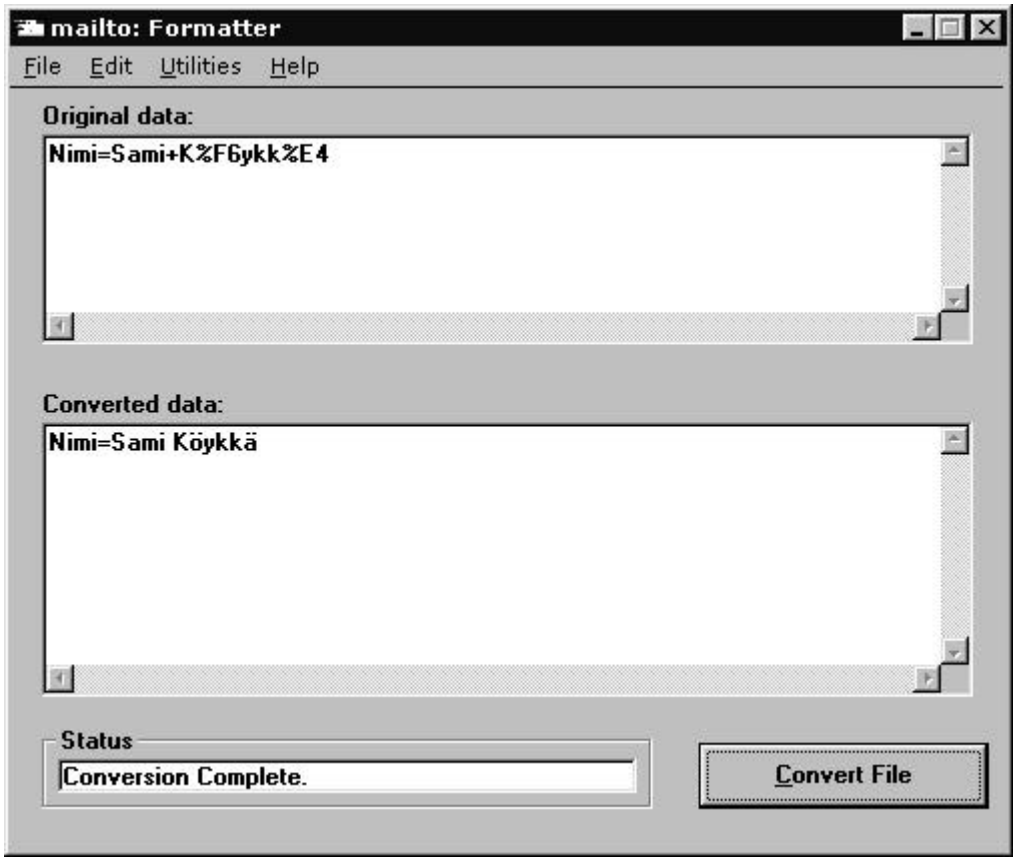
```
Nimi=Sami+K%F6ykk%E4
```

Selaimesi muotoilee lähetysvaiheessa lomakkeelta tulevan tekstin tällä tavalla: se korvaa välilyönnit plus-merkeillä ja erikoismerkit (esimerkiksi ääkköset) niitä vastaavilla merkkikoodeilla.

Voit purkaa lomakkeilta saamasi sähköpostit selväkieliseen apuohjelman avulla. Yksi tällainen apuohjelma on **mailto:formatter**, jonka voit kopioida itsellesi seuraavasta osoitteesta:

```
http://junior.apk.net/~jbarta/tutor/
forms/mtfsites.html
```

Tämän ohjelma osaa muokata sähköpostissa saamasi palautteet selväkieliseen muotoon. Voit leikata ja liimata tekstin sähköpostiohjelmastasi **formatterin** ikkunaan ja siirtää sieltä muokatut palautteet joko muihin ohjelmiin tai tiedostoihin.



Jos sijoitat lomakkeeseesi monta lomaketta, niistä jokainen tulee sähköpostiviestissä erillisenä kenttänä - ensin tekstikentän nimi ja sitten sen arvo, esimerkiksi:

```
Nimi=Matti Meikäläinen
Osoite=Hämeenkatu 1
Postitoimipaikka=33180 Tampere
```

LISÄÄ TEKSTIKENTISTÄ

Voit määritellä tekstilaatikon leveyden attribuutilla **size**. Attribuutti **maxlength** kertoo, miten monta merkkiä tekstilaatikkoon voi kirjoittaa. Jos tekstin maksimipituus on suurempi kuin kuvaruudulla näkyvä osa, selain vierittää tekstiä piiloon sitä mukaa kun käyttäjä kirjoittaa uutta. Esimerkiksi:

```
<input type=text name="Postinumero" size=5
maxlength=5>
```

Input-komennon tyyppi voi olla myös **password**. Tällainen tekstikenttä toimii muuten samalla tavalla kuin tavallinen tekstilaatikko, mutta siihen kirjoitettava teksti tulee näkyviin selaimen ruudulla tähtimerkkeinä. Voit käyttää tätä niissä tilanteissa, joissa pyydät lukijaltasi salaista tietoa (jota hän ei halua näyttää mahdollisille olkapään yli kurkistelijoille).

Voit myös kirjoittaa tekstilaatikkoihin oletusarvoisen sisällön attribuutilla **value**:

```
<input type=text name="Vuosi" value="1997">
```

Oletusarvoinen sisältö tulee näkyviin laatikkoon valmiiksi kirjoitettuna silloin, kun lukija saapuu sivulle. Lukija voi muokata laatikossa olevaa tekstiä haluamakseen, mutta jos hän ei koske laatikon sisältöön sen oletusarvoinen sisältö lähtee matkaan muun palautteen mukana.

Tekstialueen avulla sivulla voi esittää useasta rivistä koostuvan tekstinsyöttöalueen. Tekstialueet merkitään komennolla **textarea**, johon lisätään attribuuteilla **rows** ja **cols** sen koko riveinä ja sarakkeina.

Tekstialue sisältää aloitus- ja lopetuskomennon. Niiden väliin jäävälle alueelle voi kirjoittaa tekstiä, joka tulee näkyviin tekstialueelle valmiiksi kirjoitettuna silloin, kun sivu avautuu lukijan näytölle. Lukija voi muokata tekstiä haluamakseen. Jos aloitus- ja lopetuskomennon väliin ei kirjoiteta tekstiä, tekstialue tulee näkyviin tyhjänä.

Esimerkkinä on lomakkeessa sijaitseva 40 x 10 merkin kokoinen tekstialue, jossa kysytään lukijan kommentteja. Tekstialueelle laitetaan oletusarvoksi teksti, joka tulee lukijalle näkyviin valmiiksi kirjoitettuna. Lukija voi kirjoittaa omat kommenttinsa joko valmiin tekstin perään tai hän voi poistaa oletusarvoisen tekstin ja kirjoittaa omat kommenttinsa tyhjälle alueelle.

```
<textarea name="kommentit" rows=10 cols=40>
Mielipiteeni on:
</textarea>
```

Tekstialueelle kirjoitettu tieto tulee näkyviin samalla tavalla kuin tekstilaatikoihinkin kirjoitettu tieto. Tekstialueen tieto saattaa kuitenkin sisältää useita rivejä tekstiä. Esimerkiksi:

```
kommentit=Mielipiteeni on: Tämä
tekstilaatikko on erittäin huvittavan näköinen.
```

Jos haluat välittää lomakkeelta ohjelmallesi muutakin kuin käyttäjän syöttämää tietoa, voit käyttää tähän tarkoitukseen piilotettuja kenttiä. Nämä kentät eivät näy käyttäjälle eikä niiden sisältöä voi muuttaa, mutta niiden sisältämät tiedot tulevat palautteen mukana käsiteltäviksi samalla tavalla kuin tekstilaatikoidenkin.

Piilotettujen kenttien tyyppiattribuutti on **hidden**. Esimerkki:

```
<input type=hidden name="Lomake"
value="Mielipidetiedustelu">
```

Lomaketta käsittelevä ohjelma saa syötettyjen tietojen joukossa myös nimi-sisältö-parin:

```
Lomake=Mielipidetiedustelu
```

PAINIKKEET

Lomakkeisiin on mahdollista sijoittaa kahdenlaisia painikkeita: radiopainikkeita ja "rasti ruutuun" –painikkeita. Radiopainikkeet toimivat kuin muinaisten radioiden tai televisioiden kanavanvalitsimet: samaan joukkoon kuuluvista painikkeista vain yksi voi olla painettuna alas. Jos joku toinen radiopainike painetaan pohjaan, edellinen nousee ylös.

Rasti ruutuun –painikkeet taas ovat nimensä mukaisia: ruudulla olevista painikkeista voi valita ("rastittaa") yhden, useampia tai ei yhtään.

Painikkeet lisätään sivulle **input**-komennolla. Aikaisemmasta, tekstilaatikon toteuttavasta input-komennosta poiketen painikkeen komennossa tyyppi on joko **radio** tai **checkbox**. Normaalisti kaikki painikkeet ovat valitsemattomia, kun sivu ilmestyy näytölle. Jos kuitenkin haluat, että joku painike on valmiiksi valittu (rastitettu tai pohjaan painettu), voit lisätä komentoon attribuutin **checked**. Sivun lukija voi tietysti muuttaa valintoja mielensä mukaan.

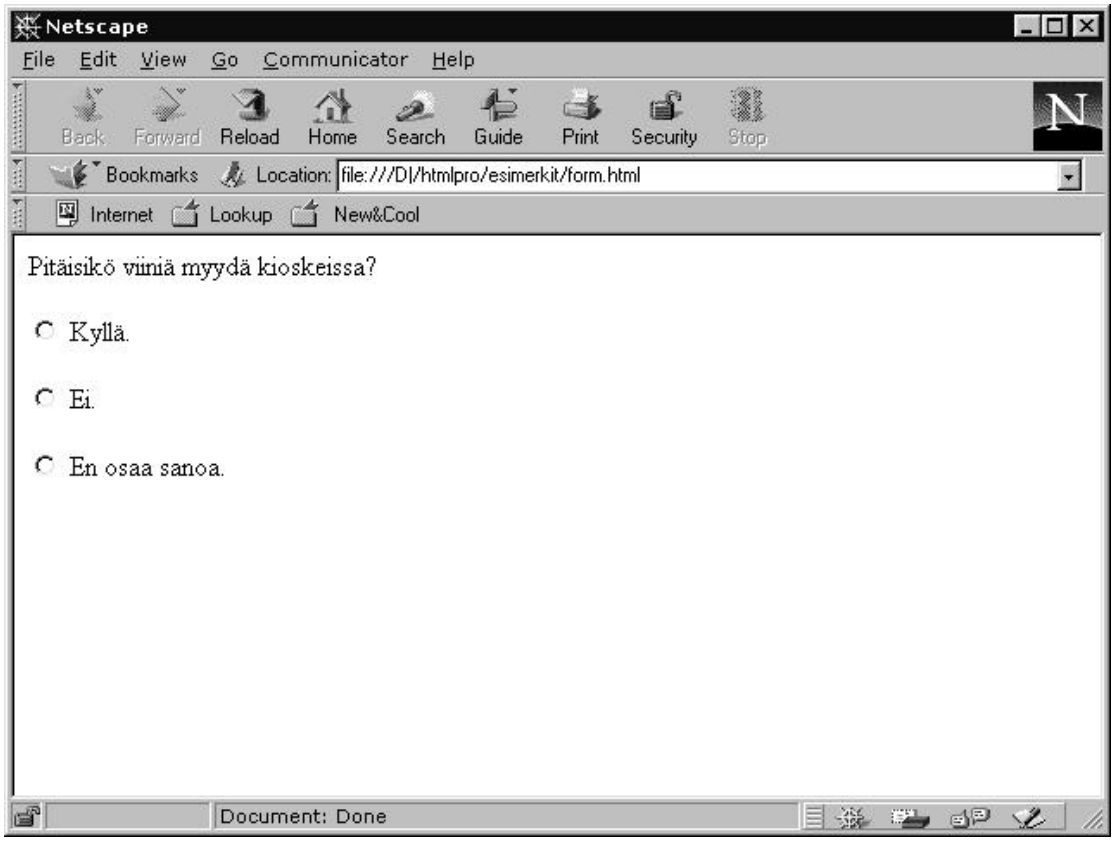
Kuten muutkin lomakekomennot, myös painikekomennot tekevät lomakkeeseen pelkän painikkeen. Jos haluat lisätä niihin myös selitystekstin, tee se normaalilla leipätekstillä tai muulla haluamallasi tyylillä.

Kun lomake lähetetään käsiteltäväksi, painikkeet palauttavat tuttuun tyyliin nimi-arvo –pareja. Nimi tulee painikkeen **name**-attribuutista ja oletusarvo tulee attribuutista **value**. Radiopainikkeissa attribuutti **name** sisältää lisäksi tiedon samaan ryhmään kuuluvista painikkeista: kaikki painikkeet, joiden **name**-attribuutissa on sama teksti, kuuluvat samaan ryhmään eikä niistä voi olla valittuna kuin yksi kerrallaan.

Esimerkkinä lomakkeessa olevista painikekomennoista voisi olla esimerkiksi mielipidetiedustelu. Tiedustelussa esitetään kysymys — esimerkiksi "Pitäisikö viiniä myydä kioskeissa?" ja siihen tarjotaan vastaukseksi joko "Kyllä", "Ei" tai "En osaa sanoa". Vastaukset toteutetaan radiopainikkeina: vastaaja voi valita vain yhden tarjotuista vaihtoehdoista. Kyselyn toteutus voisi näyttää tältä:

```
<p>Pitäisikö viiniä myydä kioskeissa?
<p><input type=radio name="Viinikysely" value="Kyllä">
Kyllä.
<p><input type=radio name="Viinikysely" value="Ei">
Ei.
<p><input type=radio name="Viinikysely" value="EOS">
En osaa sanoa.
```

Kuvaruudulla näkyvällä sivulla on kolme radiopainiketta:



Kuvaruudun kolmessa painikkeessa on sama **name**-attribuutti ("Viinikysely"), joten ne kuuluvat siis samaan ryhmään toistensa kanssa. Kun sivun lukija näpäyttää jotain painikkeista, se tulee valituksi ja edellisenä valittu painike muuttuu jälleen valitsemattomaksi. Kun lomake lähetetään, valittu radiopainike lähettää lomakkeen tietoja käsittelevälle ohjelmalle nimensä, yhtäsuuruusmerkin (=) ja arvonsa. Esimerkiksi, jos lukija valitsi kohdan "En osaa sanoa", esimerkkilomakkeesta eteenpäin lähtevä teksti on:

```
Viinikysely=EOS
```

Valitsematta jääneistä painikkeista ei välity eteenpäin minkäänlaista tekstiä.

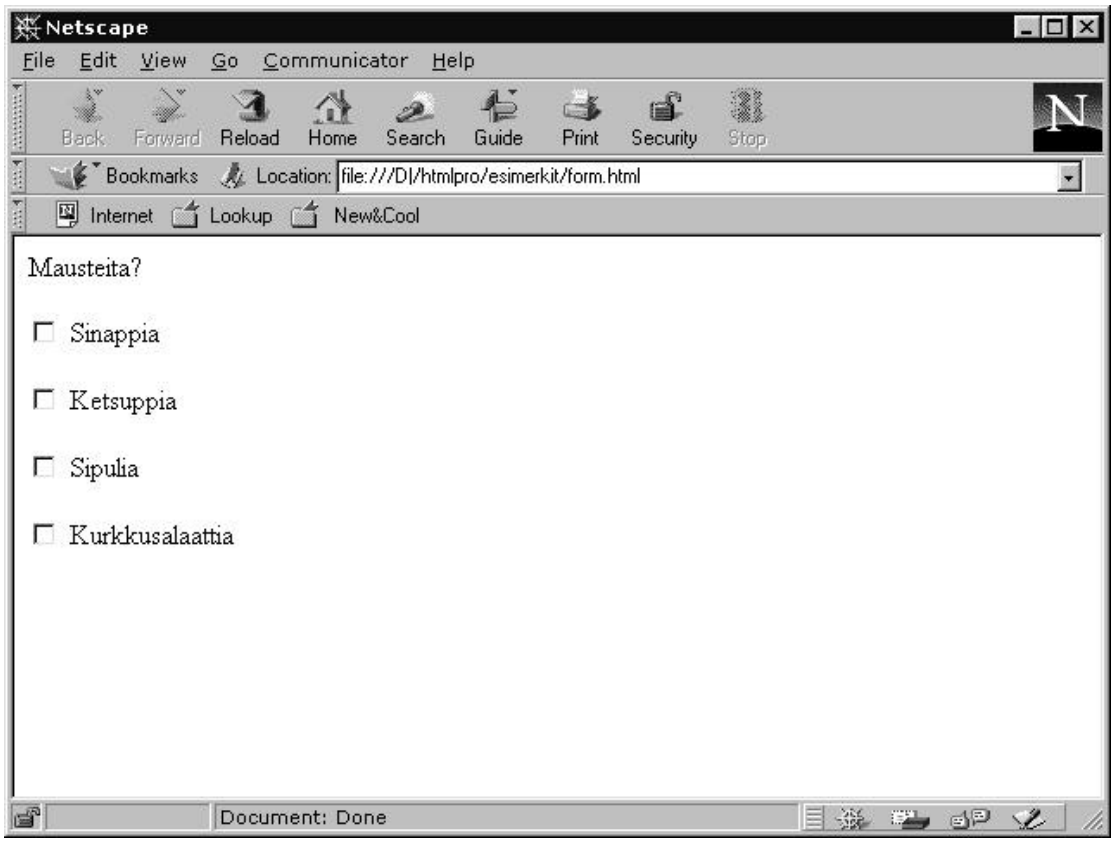
Rasti ruutuun –painikkeet määritellään samalla tavalla kuin radiopainikkeetkin. Erona on ainoastaan tyyppiattribuutti, joka on rastipainikkeissa **checkbox**. Samannimisistä rastipainikkeista voi olla valittuna useampia painikkeita yhtäaikaa.

Esimerkki rasti ruutuun –painikkeista voisi olla vaikkapa aikaansa seuraavan grillikioskin tilauslomake. Esimerkkigrillimme ottaa vastaan tilauksia

WWW:n kautta ja toimittaa tilatut ruuat kotiovelle. Yhtenä tilauslomakkeen osana on kohta, jossa kysytään tilatun ruuan mausteita. Vaihtoehtoja ovat sinappi, ketsuppi, kurkkusalaatti ja sipuli. Tilaaja voi valita näistä kaikki, joitain tai ei mitään. Lomakkeen HTML-koodi näyttää seuraavanlaiselta:

```
<p>Mausteita?
<p><input type=checkbox name="Mausteet"
value="sinappi"> Sinappia
<p><input type=checkbox name="Mausteet"
value="ketsuppi"> Ketsuppia
<p><input type=checkbox name="Mausteet"
value="sipuli"> Sipulia
<p><input type=checkbox name="Mausteet"
value="kurkkusalaatti"> Kurkkusalaattia
```

Kuvaruudulle tulee näkyviin neljä ruutua, jotka sivun lukija voi rastittaa hiirellä:



Aivan kuten radiopainikkeet, myös rastipainikkeet palauttavat ohjelmalle nimi-oletusarvo –parin. Ensin tulee nimi, sitten yhtäsuuruusmerkki ja lopuksi valitun ruudun oletusarvo. Jos edellisessä esimerkissä tilaaja olisi

ottanut nakkeihinsa sinappia ja sipulia, teksti olisi tullut sähköpostipalautteessa muodossa

```
Mausteet=sinappi
Mausteet=sipuli
```

Rasti ruutuun –painikkeissa ei tarvitse välttämättä määritellä oletusarvoa (**value**). Tiedon voi sijoittaa myös painikkeen nimeen ja oletusarvon voi jättää tyhjäksi:

```
<input type=checkbox name="lerppu">
<input type=checkbox name="korppu">
<input type=checkbox name="CD-ROM">
```

Tällöin valitun rastipainikkeen kohdalla palautetaan arvo "ON". Jos äskeisistä painikkeista valittaisiin korppu ja CD-ROM, ne palauttaisivat eteenpäin tekstin:

```
korppu=ON
CD-ROM=ON
```

Valitsemattomista painikkeista ei välity eteenpäin mitään tekstiä.

Sekä radio- että rasti ruutuun –painikkeisiin voi määritellä jotkut painikkeet valmiiksi valituiksi. Kun sivun lukijalle avautuu eteen lomake, joku radiopainikkeista saattaa olla valmiiksi alhaalla tai jossain ruudussa voi olla jo rasti. Esimerkiksi:

```
<p><input type=checkbox checked name="lahja"> Kyllä
kiitos! Lähettäkää minulle ilmainen tutustumislahja,
joka ei velvoita minua mitenkään.</p>
```

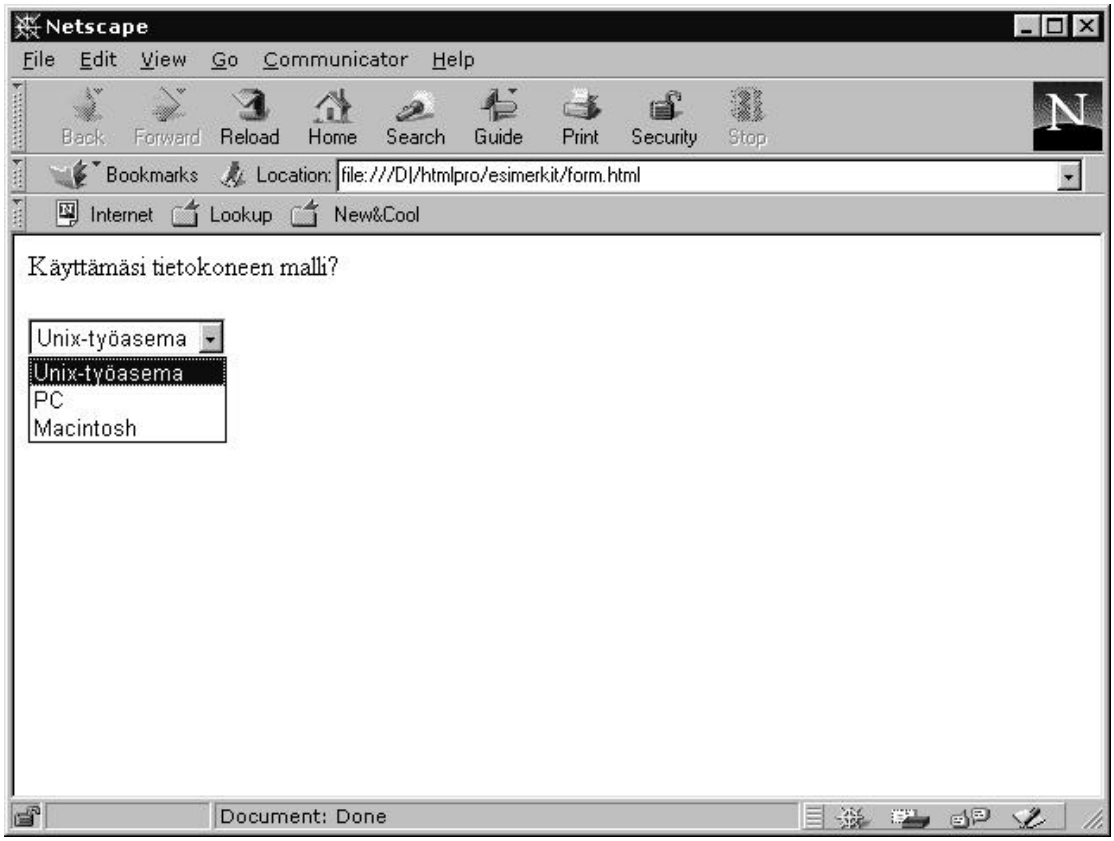
VALIKOT

Lomakkeeseen lisättävät valikot näyttävät samantapaisilta kuin muista käyttämistäsi ohjelmista tutut valikot (esimerkiksi Windowsin valikot). Valikko näkyy ruudulla tekstipalkkina, jossa on näkyvissä yksi valikon kohdista.

Valikko aloitetaan komennolla **<select>** ja päätetään komentoon **</select>**. Näiden väliin sijoitetaan **option**-komentoja, jotka kertovat valikossa valittavana olevat kohdat. Esimerkiksi yksinkertainen valikko, josta valitaan käytettävän tietokoneen malli näyttää HTML-sivuna seuraavanlaiselta:

```
<p>Käyttämäsi tietokoneen malli?</p>
<select name="Tietokonemalli">
<option>Unix-työasema
<option>PC
<option>Macintosh
</select>
```

Valikko tulee näkyviin selaimeen seuraavan näköisenä (kuvassa avattu valikko):



Kun sivun lukija valitsee jonkin valikon kohdista, se lähetetään eteenpäin lomaketta käsittelevälle ohjelmalle nimi-arvo –parina. Esimerkiksi näin:

```
Tietokonemalli=Unix-työasema
```

Yhtäsuuruusmerkin jälkeen tuleva teksti on peräisin valitun kohdan tekstistä. Jos haluat, että ohjelmalle lähtee käsiteltäväksi eri teksti kuin sivun lukija näkee ruudullaan, voit kertoa sen attribuutilla **value**:

```
<select name="maa">
<option value="fi">Suomi
<option value="se">Ruotsi
<option value="no">Norja
<option value="dk">Tanska
</select>
```

Kun sivun lukija valitsee valikosta kohdan "Suomi", ohjelmalle lähtevä tieto näyttää tällaiselta:

```
maa=fi
```

Valikossa on aina yksi kohta valittuna jo silloin, kun se tulee näkyviin lukijan näytölle. Jos et ole määritellyt tätä kohtaa erikseen, se on valikon kohdista ensimmäinen. Voit määritellä valittavaksi jonkin toisen kohdan attribuutilla **selected**:

```
<select name="Mieliväri">
<option>Punainen
<option>Keltainen
<option>Sininen
<option>Vihreä
<option selected>Musta
</select>
```

Näytölle tulevassa valikossa on valmiiksi valittuna kohta "Musta". Lukija voi tietysti halutessaan valita jonkin toisen kohdan.

Jos lisäät valikon aloituskomentoon attribuutin **multiple**, valikosta voi valita useampia kohtia. Useamman kohdan valitsemistapa riippuu käytössä olevasta selaimesta — useimmiten monta valikon kohtaa valitaan pitämällä pohjassa **Ctrl**-näppäintä ja napsauttamalla haluttuja valikon kohtia.

HTML-VERSIOT

Olet varmasti huomannut, että HTML-kieli sisältää komentoja, jotka eivät toimi kaikissa selaimissa. Uusia komentoja kehitellään jatkuvasti. Kehitystyötä tekevät sekä selainten valmistajat että yhteisö nimeltään W3C (**http://www.w3.org**).

HTML-kielestä on olemassa standardiehdotuksia. Ne luettelevat komennot, joiden voi olettaa toimivan samalla tavalla useimmissa selaimissa. Tällä hetkellä WWW-sivujen tekemiseen suositellaan standardia, jolle on annettu versionumero 3.2.

HTML 3.2 sisältää tärkeimmät peruskomennot, jotka ovat tuttuja tämänkin kirjan sivuilta. Selainvalmistajat Netscape ja Microsoft ovat kehitelleet lisäksi omia komentojaan, jotka toimivat yleensä joko ainoastaan tietyssä selainmerkissä tai sitten useimmissa tunnetuissa selaimissa. Tässä kirjassa on esitelty joitain selainkohtaisia komentoja, jotka toimivat yleensä sekä Netscapessa että Internet Explorerissa. Komentojen toiminnasta on mainittu yleensä komennon yhteydessä.

HTML:n kehitystyö jatkuu koko ajan - seuraava suunnitteilla oleva standardi on HTML 4.0, joka tulee sisältämään jälleen monia uusia käyttökelpoisia komentoja. Jotkut näistä komennoista, esimerkiksi kehykset, ovat jo nyt laajalti käytössä nykyisissä selaimissa.

Lisätietoa HTML-standardeista ja tulevista laajennuksista löydät W3C:n sivuilta osoitteesta

http://www.w3.org

SIVUKOKONAISUUKSIEN SUUNNITTELU

Seuraavassa tutustutaan sivukokonaisuuksien rakenteen ja käyttöliittymän suunnitteluun. Tutustut myös sivukokonaisuuksien ylläpitoon.

WEB-SIVUJEN TARKOITUS

Kun alat suunnitella suurempaa sivukokonaisuutta, mieti aluksi sivujesi tarkoitusta ja toimintaa. Kenelle sivusi on tarkoitettu? Mikä on sivujesi tarkoitus? Mitä pyrit saamaan aikaan sivuillasi?

Kun tiedät jo aluksi tarkasti sivujesi tarkoituksen, sivujesi suunnittelu ja myöhempi laajentaminen sujuu selkeämmin. Samalla voit analysoida tehokkaammin sitä, ovatko sivusi saavuttaneet tavoitteensa.

Kun tiedät, millaista yleisöä pyrit saavuttamaan sivuillasi, voit ottaa tämän huomioon sivujasi suunnitellessasi. Sivujen ulkoasu, niillä käytetyt tekniset ratkaisut ja niiden muoto riippuvat paljon käyttäjistäsi. Nopealukuiset, paljon erikoisefektejä ja lyhyitä tekstejä sisältävät sivut ovat mainioita käyttäjille, jotka etsivät verkosta elämyksiä ja viihdettä. Suuria tietomääriä sisältävät hakuteoksenomaiset sivut taas voivat sisältää pidempiä tekstimassoja ja niiden ulkoasu voidaan jättää karummaksi.

Jos etsit mahdollisimman laajaa yleisöä, otat sivuillasi huomioon kaikenlaisia selaimia ja tietokoneita käyttävät asiakkaat ja vältät turhaa kuvitusta tai erikoisefektejä. Sen sijaan sivut, joiden tarkoituksena on luoda vaikutus uusinta teknologiaa käyttäviin surffaajiin, voivat sisältää enemmän koristeluita.

Aloitteleville käyttäjillä tärkeimpiä ominaisuuksia ovat selkeät ja yksinkertaiset navigointivalikot, kun taas kokeneet käyttäjät haluavat löytää tietoa sivuilta mahdollisimman tehokkaasti ja nopeasti.

Rakenna sivuistasi malli ennen suunnittelun aloittamista. Tällä tavalla voit pohtia sivujesi toimintaa ja rakennetta jo etukäteen eikä sinun tarvitse tehdä muutoksia valmiisiin sivuihin.

RAKENNE

Sivujesi rakenne vaikuttaa siihen, miten pitkälle sivuille saapuvat käyttäjät jaksavat lukea niitä. Tavallinen verkon käyttäjä on tottunut vain silmäilemään sivujen sisältöä ja siirtymään nopeasti seuraavalle sivulle. Huolellisella suunnittelulla voit tehdä sivuistasi sellaiset, että kävijät jaksavat uppoutua tarjoamaasi asiaan ja tutkia sitä syvällisemmin.

Aloita miettimällä sivujesi tarkoitusta. Jaa sivusi hierarkkisiksi kokonaisuuksiksi, niin että sivuilla ei ole liian vaikeaa navigoida, mutta niitä on kuitenkin miellyttävä lukea.

Sivujen rakenne on hyvin tärkeää myös siksi, että WWW:ssä palvelimelta toiselle siirtyminen on äärimmäisen helppoa. Samalla hetkellä kun käyttäjä turhautuu käyttämään sivujasi tai kun odotusaika käy liian suureksi, todennäköisyys sille, että hän jatkaa matkaa johonkin toiseen paikkaan kasvaa voimakkaasti. Sen vuoksi sivujen suunnitteluun kannattaa uhrata paljon aikaa ja vaivaa.

NAVIGOINTI

Sivujen välinen navigointi on mahdollista toteuttaa monella eri tavalla. Navigoinnin toteutustapa vaikuttaa siihen, miten käyttäjä hahmottaa sivusi ja miten hän osaa liikkua eteenpäin.

Voit saada sivuillesi suurenkin yleisön mainostamalla niitä ahkerasti ja saamalla siihen linkkejä ympäri maailmaa, mutta jos käyttäjät käyvät sivuillasi vain kerran lukemassa sivun tai kaksi eivätkä palaa sinne enää toiste, sivujesi tarkoitus on epäonnistunut. Jos mittaat sivujesi onnistuneisuutta käyttäjämäärillä, WWW-sivujen tärkeimpiä asiakkaita ovat ne, jotka palaavat sivuillesi yhä uudelleen. Tyytyväiset kävijät kertovat sivuistasi tutuilleen ja saavat heidätkin vierailemaan sivuillasi. Tähän kaikkeen vaikuttaa yhtenä tärkeimpänä osana navigointi - jos sivuillasi käynti on miellyttävä kokemus, niiden suosio on taattu.

Kun rakennat palvelua, yksi ensimmäisistä tehtävistä on miettiä sen rakennetta. Koska et voi sijoittaa enää kaikkea tietoa yhdelle sivulle, palvelu täytyy jakaa paloiksi ja yhdistää nämä palaset toisiinsa selkeällä tavalla.

PERÄKKÄINEN RAKENNE

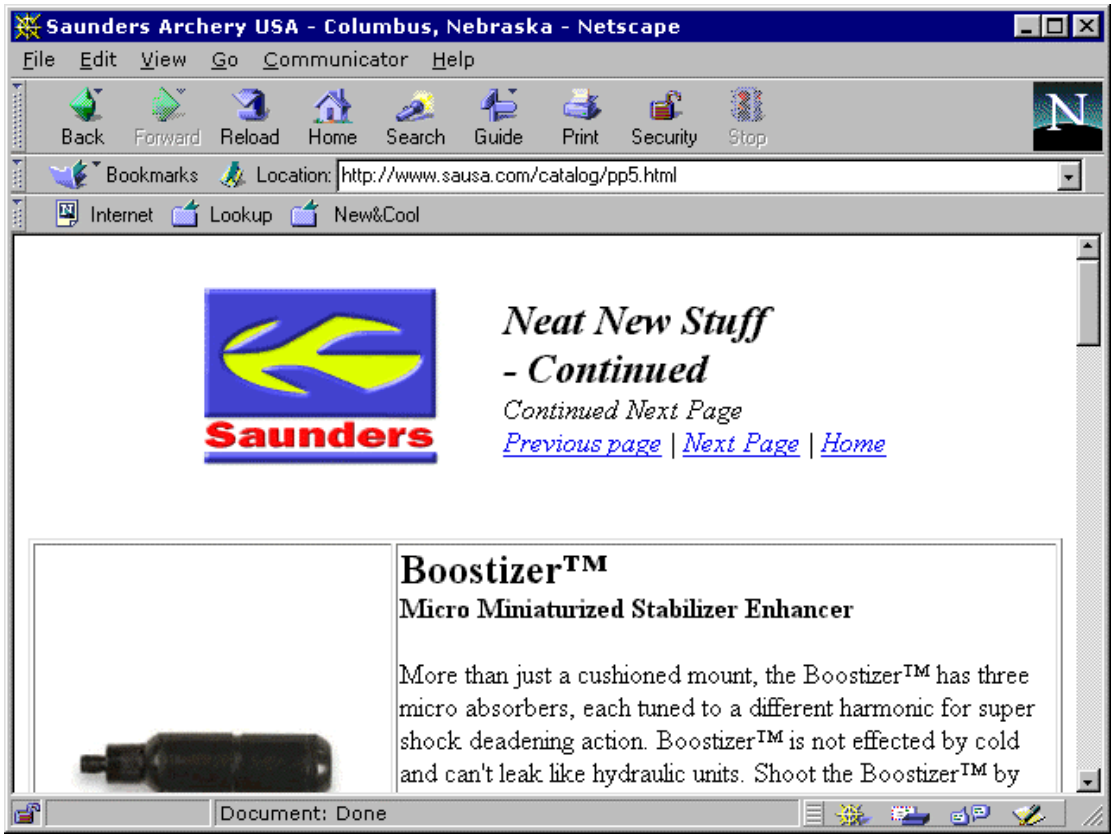
Peräkkäisistä sivuista koostuva rakenne on kaikista yksinkertaisin. Sivut sijaitsevat "peräkkäin" ja niitä voidaan lukea samalla periaatteella kuin kirjaa: Käyttäjä aloittaa ensimmäiseltä sivulta, luettuaan sen siirtyy toiselle sivulle ja niin edelleen, kunnes kaikki sivut on luettu.

Peräkkäiset sivut sisältävät myös jonkinlaisen sisällysluettelon - listan jokaisen sivun sisältämästä asiasta, jolloin käyttäjä voi myös hypätä suoraan haluamalleen sivulle. Jos käyttäjä haluaa tutkia jotain toista sivua, hän voi siirtyä seuraavalle tai edelliselle sivulle, tai takaisin sisällysluetteloon ja sitä kautta uudelle sivulle listassa.

Esimerkki peräkkäisestä rakenteesta voisi olla esimerkiksi ison palvelimen kotisivuluettelo. Luettelo sisältää linkkejä kyseisen palvelimen käyttäjien kotisivuille.

Tämä lista on aina mahdollista toteuttaa yhtenä isona listana, mutta sen lataaminen ja selailu olisi mahdotonta suurimmalle osalle käyttäjistä. Tavallinen tapa on sen sijaan jakaa lista peräkkäisiksi sivuiksi, vaikkapa käyttäjän sukunimen mukaan. Tällöin lista jakautuu useisiin sivuihin, joista ensimmäisellä on linkkejä kotisivuille, joiden tekijän sukunimi alkaa kirjaimella A, seuraavalla kirjaimella B alkaville sukunimille ja niin edelleen.

Peräkkäistä rakennetta voi käyttää myös sivujen sisällä yksittäisen asian kuvaamiseen - esimerkiksi verkkolehdessä yksi artikkeli voi jakautua useammaksi sivuksi, jotka käyttäjä lukee peräjälkeen.



Peräkkäisten sivujen navigointi on toteutettu yleensä yksinkertaisella sisällysluettelolla (kuten yllä mainittiin) sekä sivulta toiselle johtavilla edellinen- ja seuraava sivu -linkeillä.

Peräkkäisten sivujen etuna on lehdistä ja kirjoista tuttu rakenne. Käyttäjä pystyy helposti rakentamaan mielessään mallin siitä, miltä sivujen rakenne näyttää ja hänellä on sillä tavalla selkeä käsitys siitä, mitä hän voi tehdä ja minne hän voi mennä.

Peräkkäinen rakenne yksinään ei sovi kuitenkaan kovin laajaan tietomäärään.

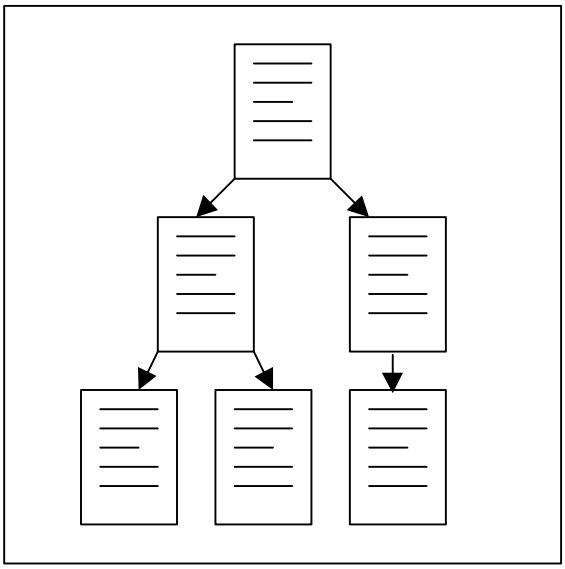
HIERARKKINEN RAKENNE

Hierarkkisessa mallissa sivujen sisältö jaetaan puumaiseksi rakenteeksi. Sivuille saavutaan etusivulta (puun juuresta), josta on mahdollisuus valita kulkutieksi jokin tarjolla olevista aiheista (puun oksat).

Kun käyttäjä siirtyy tiettyyn aiheeseen, hän kohtaa uusia mahdollisuuksia siirtyä asiassa eteenpäin, samalla tavalla kuin puun oksa jakautuu pienemmiksi oksiksi ja lopulta lehdiksi.

Hierarkkinen malli sopii erityisen hyvin WWW-palvelimien rakenteen kuvaajaksi, koska yleensä palvelimissakin on tietty etusivu, josta useimmat käyttäjät lähtevät liikkeelle.

Useimmat käyttäjät ovat tuttuja hierarkkisen mallin kanssa, joten siinä navigointi sujuu helposti.



Jokaisesta hierarkkisen mallin alla olevasta sivusta voidaan tehdä linkkejä sekä palvelimen etusivulle että myös senhetkisen aiheen pääsivulle (lähimpään oksanhaaraan). Lisäksi sivut voivat sisältää linkkejä seuraaviin ja edellisiin aiheisiin ja ehkäpä peräti seuraaviin ja edellisiin sivuihin. Ylenmääräisten linkkien tekemistä kannattaa kuitenkin varoa, etteivät käyttäjät sekoa liian monen tarjolla olevan vaihtoehdon kanssa.

Toinen hierarkkisessa rakenteessa piilevä vaara on tehdä rakenteesta liian monitasoinen, jolloin käyttäjä joutuu varsinaiseen asiaan päästäkseen kahlaamaan useiden valikoiden läpi. Liian syvästä hierarkkisesta rakenteesta saattaa jäädä päällimmäiseksi mielikuvaksi jatkuva navigoinnin tarve, joka saattaa pelottaa käyttäjät pois palvelusta.

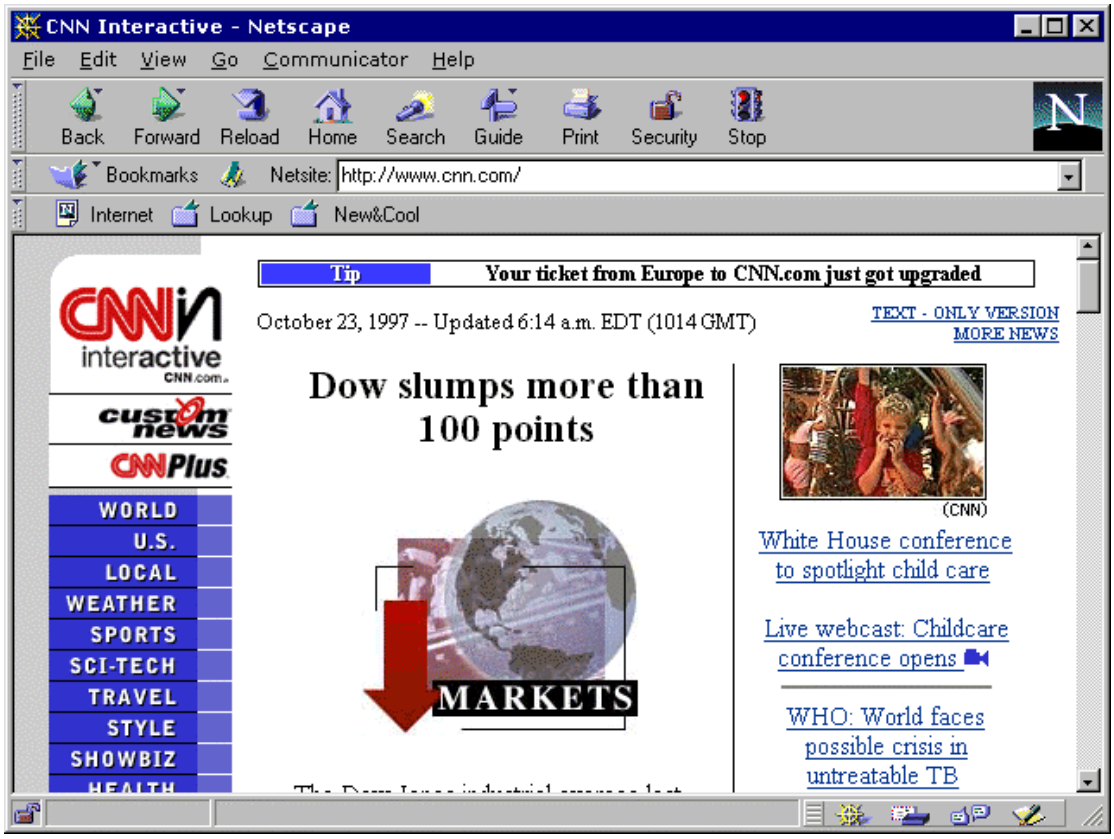
Liian syvää hierarkkista rakennetta voi madaltaa leventämällä sitä, jolloin rakenne sisältää vähemmän haarautumia ja oksia syvemmällä tasolla. Käyttäjä pääsee nopeammin käsiksi varsinaiseen asiaan. Tässä on kuitenkin vaarana se, että rakenne muodostuu liian leveäksi, jolloin haarautumakohdissa on tarjolla niin monia vaihtoehtoja, että ne yksinään lannistavat käyttäjän.

Toimivan hierarkkisen rakenteen tekeminen onkin jatkuvaa tasapainoilua syvyyden ja leveyden välillä. Peukalosääntönä voisi pitää viiden rajoitusta: hierarkian tulisi jakautua viiteen aihepiiriin ja olla syvyydeltään korkeintaan viisi tasoa (käyttäjän ei tarvitse napsauttaa viittä kertaa enempää päästäkseen haluamaansa tietoon käsiksi). Näitä rajoja voi tietysti venyttää palvelimen suuruuden ja aiheiden monimuotoisuuden mukaan.

Kolmas tapa rajoittaa hierarkian sekavuutta on lisätä rakenteeseen myös muunlaisia alueita - varsinainen palvelimessa oleva tieto voidaan järjestää

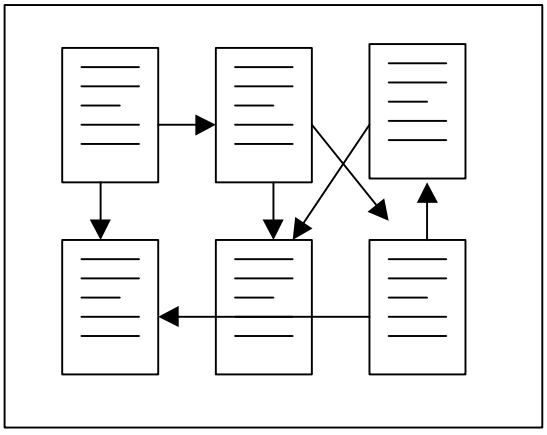
peräkkäisiksi sivuiksi ja pelkkä navigointi voidaan toteuttaa hierarkkisella tavalla.

Esimerkkejä hierarkkisista malleista on nähtävissä monissa WWW-palvelimissa - katso vaikkapa CNN:n uutispalvelinta osoitteessa **`www.cnn.com`**, jossa uutiset on jaettu etusivun uutisten lisäksi kansainvälisiin, USA:laisiin ja paikallisuutisiin, säähän, urheiluun jne.



VERKKORAKENNE

Verkkomalli ei sisällä varsinaisesti ollenkaan navigointirakennetta vaan siinä sivut ovat linkitettyinä toisiinsa tekstin sisällä olevista linkeistä. Tällaisessa rakenteessa ideana on saada aikaan vapaasti liikkuvien ideoiden muodostama verkko, jossa käyttäjät voivat siirtyä sivulta toiselle vapaasti omien mielleyhtymiensä mukaan.



Tällaisessa tapauksessa palvelimen rakenne näkyy jokaiselle sinne saapuvalle käyttäjälle hieman erilaisena - jokainen käyttäjä suunnistaa palvelun sisällä itse

omien mieltymystensä mukaisesti eikä palvelun suunnittelija ole rajoittanut hänen navigointiaan keinotekoisilla rajoituksilla.

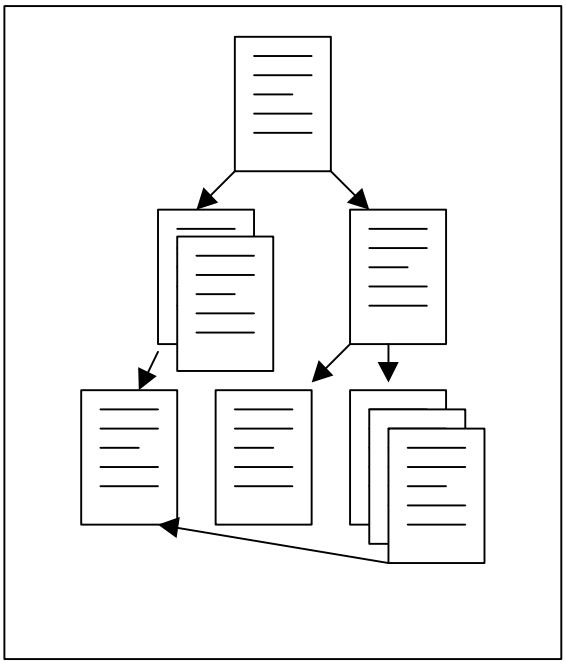
Verkkomallinen navigointi vaatii, että palvelimen sisältö sisältää paljon linkkejä - mitä useammasta sanasta tai navigointipainikkeesta pääsee eteenpäin relevanteille sivuille, sitä parempi.

Verkkomallinen navigointi sopii mainiosti pienille sivustoille ja sellaisiin palveluihin, joissa vapaan sivuilla liikkumisen aiheuttama elämys on tärkeä.

Verkkonavigointi saattaa kuitenkin myös sekoittaa käyttäjän. Monimutkaisen verkon hahmottaminen on hyvin vaikeaa, ellei peräti mahdotonta. Jos tarkoituksenasi on tarjota käyttäjälle mahdollisuus tajuta palvelimen rakenne ja navigoida tehokkaasti juuri sinne minne hän haluaa mennä, verkkomalli ei välttämättä sovi palvelun ainoaksi rakenteeksi.

YHDISTELMÄRAKENNE

Yleisin ja tehokkain tapa rakentaa sivustoja on yhdistellä erilaisia rakenteita. Palvelimen pohjalla saattaa olla esimerkiksi hierarkkinen rakenne, jota täydennetään peräkkäisillä rakenteilla kohdissa, joihin tämä sopii parhaiten. Lisäksi käyttäjälle tarjotaan mahdollisuus liikkua sivujen välillä vapaamuotoisesti tekemällä tekstistä linkkejä asiaankuuluviin kohtiin ympäri sivuja.



Verkossa on tarjolla monia esimerkkejä tällaisesta rakenteesta: esimerkiksi **C|Netin** sivuilla (**`http://www.cnet.com`**) on tarjolla hierarkkinen tie eri aihepiireihin, jotka on jaoteltu peräkkäisiksi rakenteiksi. Lisäksi tekstin sisällä on jatkuvasti verkkotyylisiä linkkejä eri aihepiirien välillä niin että käyttäjät voivat alkaa seurata mielenkiintoista ajatusketjua näin halutessaan.

SANAHAKU

WWW-käyttäjien keskuudessa tehdyssä tutkimuksessa on todettu, että puolet sivuja selailevista käyttäjistä käyttää selailun sijasta mieluummin sanahakua. Viidennes käyttäjistä liikkuu pelkästään käyttäen tarjolla olevia linkkejä ja loppuosa käyttää sekaisin molempia.

Palvelimessa, jonka sisältö ylittää 200 sivua, tulisi olla käytössä sanahaku, jonka avulla palvelimesta voi etsiä haluamansa sivun sen sisältämien sanojen perusteella. Se puoli käyttäjistä, joka ei ensisijaisesti käytä sanahakua tietojen etsimiseen saattaa eksyä tai turhautua navigointiin sivuilla liikkuessaan. Tällaisia käyttäjiä varten palvelimen jokaisella sivulla tulisi olla mahdollisuus siirtyä sanahakuun.

Pelkkää sanahakua ei kuitenkaan kannata sijoittaa palvelimen ainoaksi hakutoiminnoksi, koska käyttäjät saattavat haluta siirtyä sanahaun tuloksena saamaltaan sivulta eteenpäin loogisessa järjestyksessä, esimerkiksi silloin kun he ovat löytäneet mielenkiintoisen aiheen ja haluavat tutkia kaiken siitä tarjolla olevan tiedon.

Yleisin, ja useimmiten myös paras tapa toteuttaa sanahaku on tarjota haettavaksi koko palvelimen sisältö kerralla. Jossain tapauksissa on myös oikeutettua tarjota haku vain tiettyyn alihaaraan palvelimen sisällössä, mutta tällaisen haun yhteydessä kannattaa mainita selkeästi haun rajoittuneisuudesta - muussa tapauksessa käyttäjät saattavat luulla etsivänsä tietoa koko palvelimesta ja turhautua kun haluttua tietoa ei löydykään.

ETUSIVU

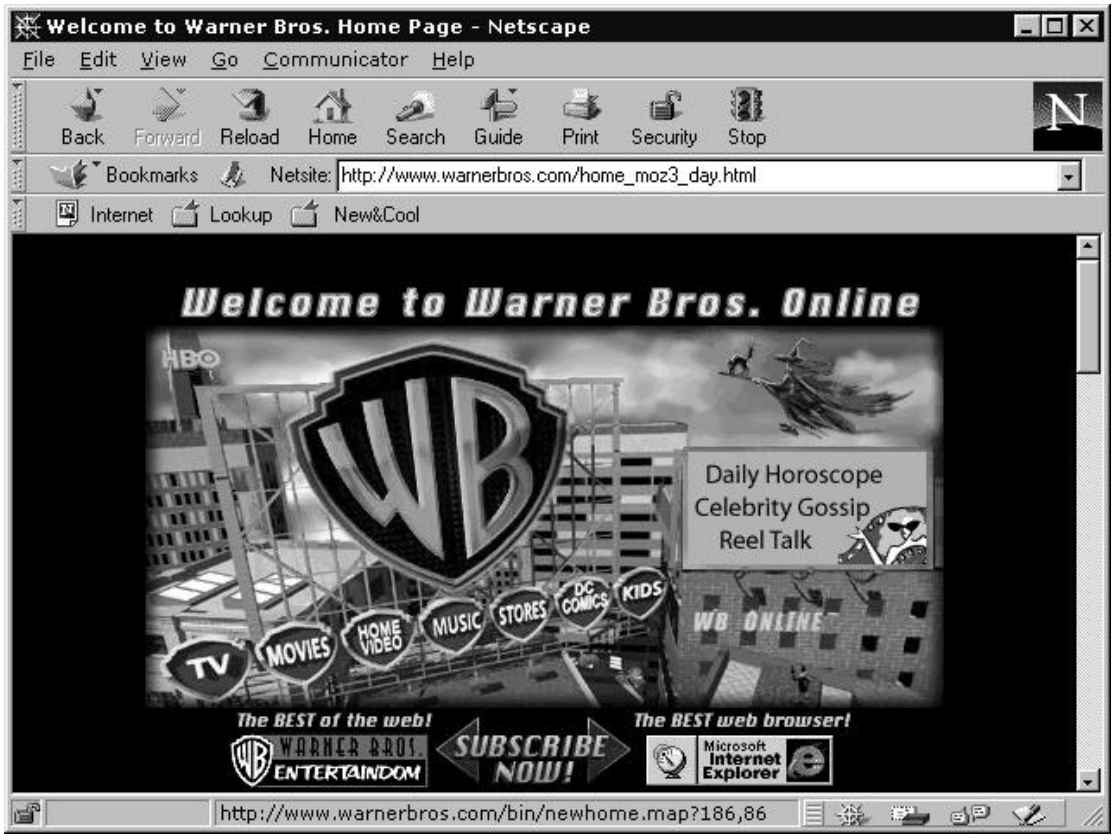
Palvelun etusivu on yksi tärkeimmistä palvelimessasi. Tämä on se sivu, jonka käyttäjät yleensä näkevät ensimmäisenä saapuessaan sivuillesi.

Kun mainostat sivujasi joko verkossa tai muissa medioissa, voit yleensä antaa vain yhden osoitteen, joka tyypillisesti osoittaa palvelusi kotisivulle.

Etusivun tulisi latautua nopeasti ja sen tulisi antaa käyttäjälle välittömästi kuva siitä, mitä palvelussa on tarjolla. Koska verkossa liikkuvat käyttäjät ovat perinteisesti hyvin kärsimättömiä, erilaisten vaihtoehtojen tulisi olla lisäksi hyvin esillä - käyttäjän tulisi voida yhdellä silmäyksellä nähdä, mitä hänelle on tarjolla ja miten hän voi liikkua sivuilta eteenpäin.

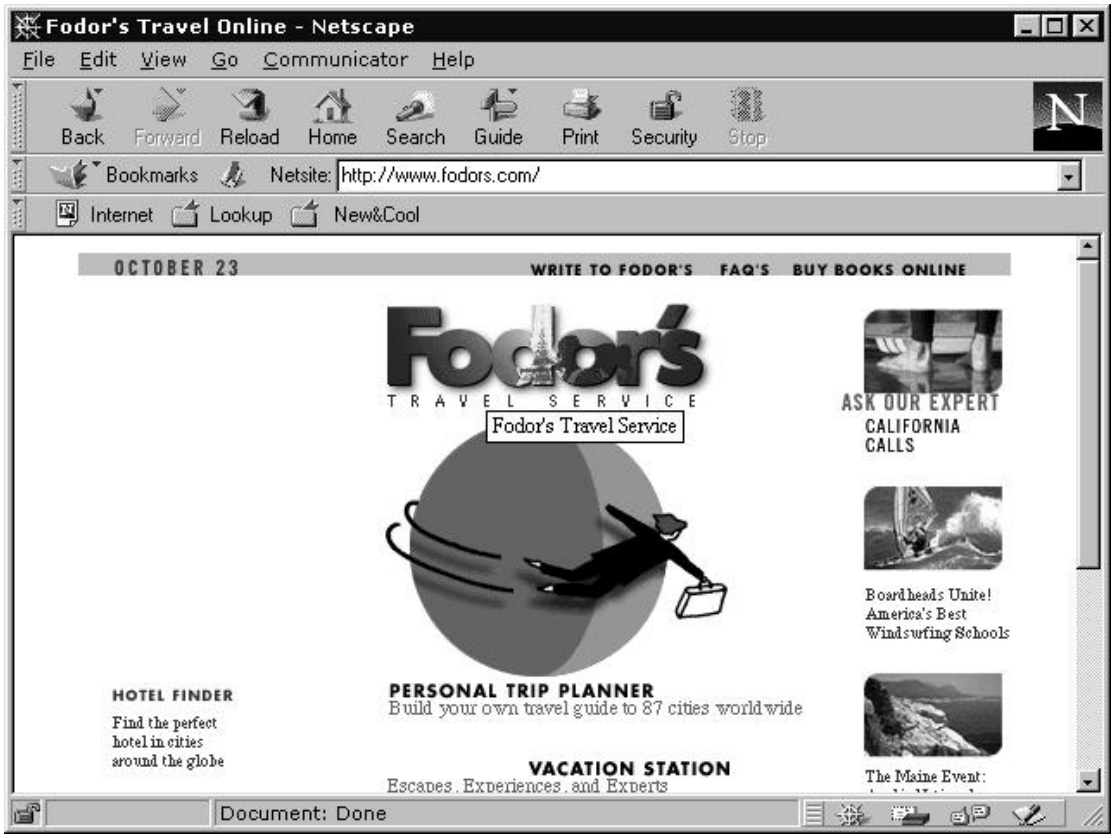
GRAAFINEN ETUSIVU

Monet palvelimet sisältävät etusivullaan suuren interaktiivisen kuvan. Tällaisen kuvan etuna on sen monipuolisuus: kuvassa voi käyttää erilaisia graafisia tehosteita, värein ja tekstityypein korostettua tekstiä ja monipuolista muotoilua. Interaktiivisesta kuvasta saa aikaan houkuttelevan näköisen.



Interaktiivisella kuvalla toteutettu etusivu tehdään piirtämällä sopiva kuva, rakentamalla sille karttamäärittelyt ja sijoittamalla etusivulle sopivat komennot. Tästä seuraa kuitenkin myös graafisen etusivun päivitysongelma. Kun etusivun kuvaa muutetaan, kuva täytyy piirtää uudelleen. Jos kuvassa olevat erilaiset aktiiviset kohdat muuttavat muotoa ja paikkaa, kuvan karttatiedosto täytyy myös luoda uudelleen.

Toinen mahdollisuus graafisen etusivun tekemiseen on rakentaa etusivu käyttäen tekstiä ja (enemmän tai vähemmän) kuvia. Tällöin etusivun ulkoasu suunnitellaan HTML-kielen mahdollisuuksien ja rajoitusten mukaan. Etusivu koostuu tekstistä, jota on elävöitetty erilaisilla kuvilla tai pelkistä kuvista, jotka on sijoitettu ympäri ruutua.



Tällaisen sivun muokkaus on helpompaa. Jos sivulla oleva teksti tai kuvien asettelu muuttuu, pelkkä HTML-tiedoston editointi riittää. Jos kuvitus muuttuu, kokonaisuudesta tarvitsee muuttaa ainoastaan niitä kuvia, joita muutos koskee.

Riippumatta siitä, käytetäänkö interaktiivista kuvaa vai tavallisia kuvia, graafisen etusivun tärkeä ominaisuus on latautumisen nopeus. Jos etusivusi sisältää suuria kuvia, sen lataus kestää kauan etenkin modeemikäyttäjillä ja kaukaa verkosta tulevilla lukijoilla. Jos etusivun lataus kestää liian kauan, osa lukijoista pysäyttää latauksen ja jatkaa johonkin toiseen paikkaan — ehkäpä kilpailijasi sivuille. Verkon ruuhka-aikoina tilanne pahenee entisestään. Etusivulla käytetyt kuvat kannattaakin optimoida tiedostokooltaan mahdollisimman pieniksi.

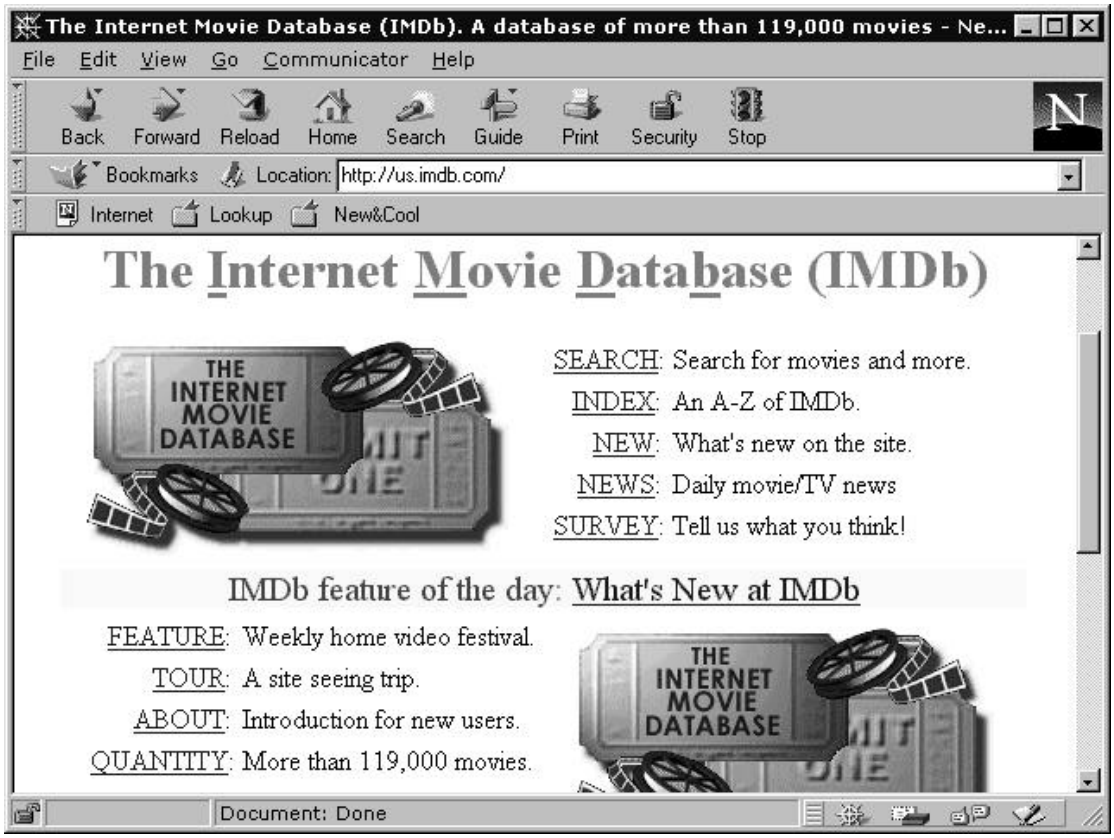
Jos suunnittelet etusivusi graafiseksi, muista kuitenkin tarjota samalla sivulla myös tekstipohjainen vaihtoehto. Vaihtoehtoa tarvitsevat ne lukijat, jotka käyttävät pelkkää tekstiä näyttävää selainta ja lukijat, jotka ovat kytkeneet kuvien latauksen pois päältä selaimessaan.

Tekstipohjainen vaihtoehto voi koostua ruudun alareunassa olevasta listasta, joka sisältää tekstilinkin jokaiseen interaktiivisen tai tavallisen kuvan kautta saatavaan sivuun. Vaihtoehto voi olla myös yksittäinen linkki,

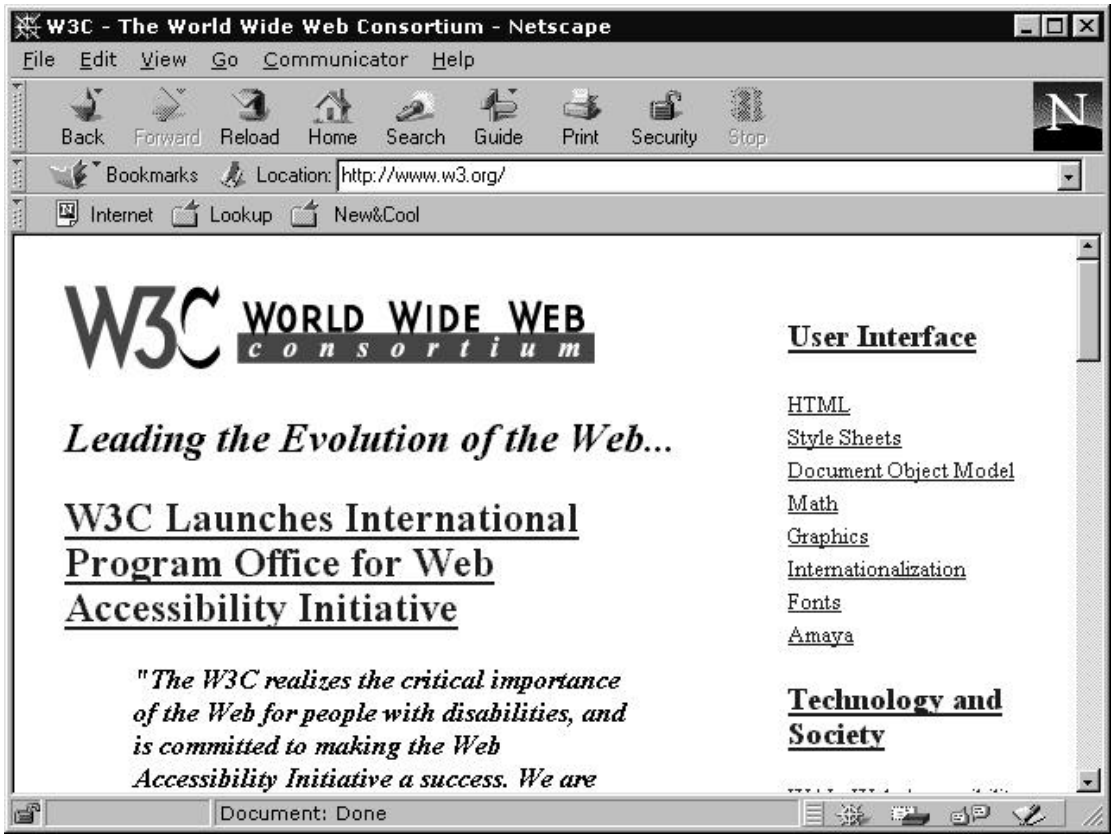
joka johtaa etusivusta tehtyyn tekstipohjaiseen versioon. Tekstipohjaisesta etusivusta tulee olla myös linkki takaisin graafiseen etusivuun niille lukijoille, jotka saapuvat sivuillesi tekstipohjaisen etusivun kautta.

TEKSTIETUSIVU

Koko palvelun etusivu voi olla myös suunniteltu tekstipohjaiseksi. Tekstipohjainen etusivu koostuu pääasiassa tai kokonaisuudessaan tekstistä. Yhteydet eteenpäin sivuille rakennetaan tavallisina linkkeinä. Myös tekstisivun elävöittäjinä saattaa olla kuvia — esimerkiksi sivukokonaisuuden logo yläreunassa ja eri aihepiirejä kuvaavat ikonit kyseiseen aihepiiriin johtavan linkin yhteydessä.



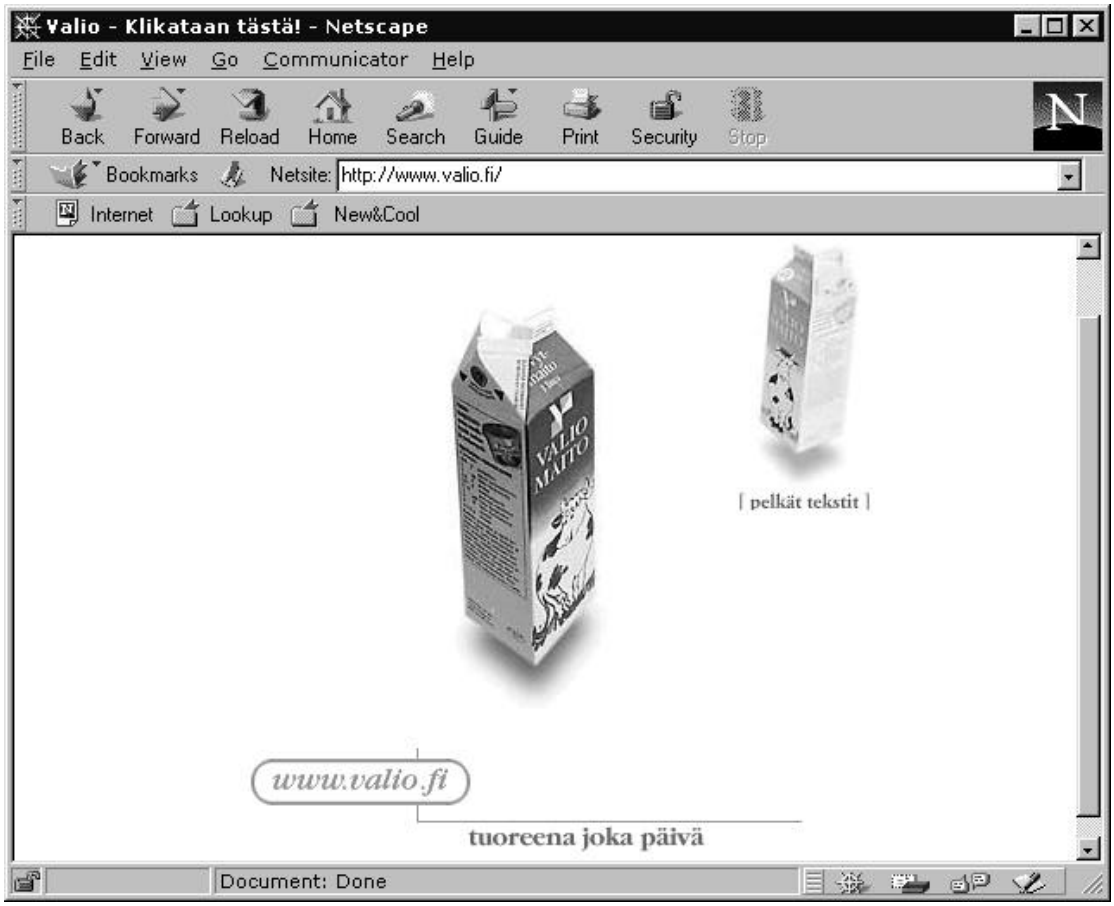
Pelkän tekstin etuna on lataamisen nopeus ja sen riippumattomuus laitteistosta. Mitä vähemmän sivu sisältää kuvia, sitä nopeammin se saapuu lukijan näytölle — etenkin, jos lukija saapuu kauempaa maailmalta tai vaikkapa modeemin välityksellä. Koska HTML-kieli on rakennettu laitteistosta riippumattomaksi, HTML-komennoilla toteutettu tekstipohjainen etusivu toimii missä tahansa selaimessa. Sitä voivat käyttää niin tekstipäätteen kuin graafisen työasemankin käyttäjät. Teksti muotoutuu automaattisesti erikokoisille näytöille ja erilaisille tekstityypeille.



Tavanomainen ulkonäkö voi pelottaa pois sellaisia lukijoita, jotka etsivät verkosta lähinnä visuaalisia elämyksiä. Jos kuitenkin suunnittelet etusivusi tekstit mukaansa tempaaviksi, palvelimesi pystyy kilpailemaan hienoimpienkin graafisten etusivujen kanssa.

TERVETULOA-SIVU

Etusivu voi olla myös eräänlainen tervetuloa-sivu. Sivu sisältää yhden tai muutamia kuvia ja linkin, joka johtaa varsinaisille sivuille. Tällaisen sivun avulla voit herättää lukijasi mielenkiinnon ja houkutella hänet syvemmälle.



Koska sivulla ei tarvita monia kuvia tai pitkiä tekstimääriä, voit tehdä siitä nopeasti latautuvan. Varsinaisen asian ja linkit voit sijoittaa seuraavalle sivulle, joka on palvelusi varsinainen kotisivu.

Tervetuloa-sivun ansiosta voit tehdä varsinaisestakin sivusta kevyemmän, koska sinne ei tarvitse enää välttämättä sijoittaa niin paljoa grafiikkaa.

Tervetulosivulle kannattaa sijoittaa komento, joka siirtää selaimen etusivulle tietyn ajan kuluttua. Tällä tavalla käyttäjä siirtyy automaattisesti varsinaiselle etusivulle siinäkin tapauksessa, että hän ei löydä tervetulosivulta linkkiä.

Automaattinen siirto tapahtuu lisäämällä sivun alkuun tietynlainen **meta**-komento:

```
<HTML>
<HEAD>
<TITLE>Tervetulosivu</TITLE>
<META HTTP-EQUIV="Refresh"
CONTENT="20; URL=index2.html">
```

```
</HEAD>
<BODY>
```

Komentoon sijoitetaan sekuntimäärä, jonka kuluttua sivua vaihdetaan ja URL-osoite (tai tiedostonnimi) sivulle, jolle siirrytään seuraavaksi.

KÄYTÄNNÖN VINKKEJÄ

Kun rakennat sivujasi, yritä välttää jo alusta pitäen eräitä yleisiä virheitä. Tällaisia ovat esimerkiksi huonosti nimetyt tai orvot sivut.

TARKISTA SIVUJEN NIMET

Kuten aiemmin kerrottiin, jokaiselle WWW-sivulle annetaan nimi sijoittamalla sivun alkuun **<title>** -komento. Sen aloitus- ja lopetuskomennon väliin sijoitetaan sivun nimi.

Muista tehdä sivujesi nimistä kuvaavia. Nimet tulevat näkyviin moniin paikkoihin, esimerkiksi käyttäjän selaimen historiavalikkoon, **Back**- ja **Forward**-painikkeista saataviin valikoihin ja kirjanmerkkeihin.

Kun sivujesi nimet ovat kuvaavia, käyttäjä pystyy tehokkaasti Siirtymään sivulta toiselle **Back**- ja **Forward**-painikkeita käyttäen. Kun käyttäjä myöhemmin tutkii selaimensa historiavalikoita, hän pystyy yhdellä silmäyksellä erottamaan listasta sinun sivukokonaisuuteesi kuuluvat sivut ja palaamaan niille halutessaan.

Jos käyttäjä päättää tehdä selaimeensa sivuillesi osoittavan kirjanmerkin, hänen kirjanmerkkinsä otsikoksi tulee automaattisesti sivusi nimi. Jos nimi on hyvin suunniteltu, käyttäjän ei tarvitse erikseen enää muokata tätä sivua.

Myös monet hakukoneet, kuten **Alta Vista**, näyttävät sivusi otsikon hakutuloksen otsikkona.

Hyvä sivun nimi kertoo katsojalle yhdellä silmäyksellä palvelun nimen, jossa sivu sijaitsee. Siitä pystyy myös näkemään, missä palvelimen osassa sivu sijaitsee. Jos haluat luoda ammattimaisen vaikutelman, voit esimerkiksi erottaa sivun nimessä palvelun nimen ja sivun sijainnin palvelun sisällä käyttämällä kaksoispistettä:

Oy Firma Ab: Yritysesittely

Älä käytä samaa nimeä kaikilla palvelusi sivuilla - tällöin käyttäjän historiavalikko saattaa näyttää tällaiselta:

Oy Firma Ab
Oy Firma Ab
Oy Firma Ab
Oy Firma Ab

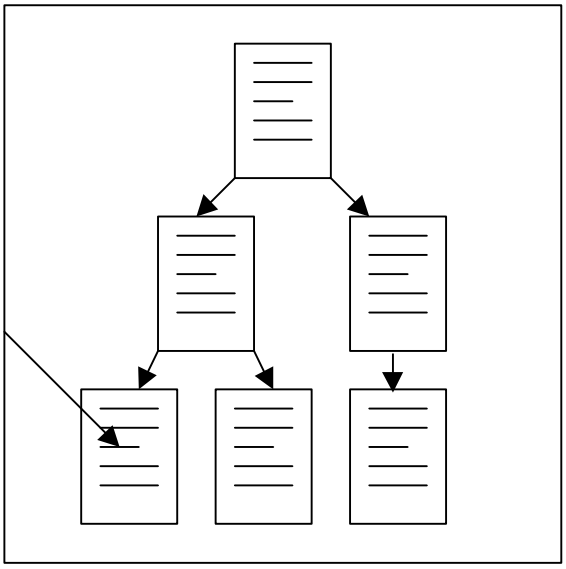
Kun olet käyttänyt jokaisella sivulla kuvaavaa sivun nimeä, käyttäjä saattaakin nähdä **Back**-valikkoa tai historiavalikkoa selatessaan tällaisen listan:

Oy Firma Ab: Biotekniikan laboratorio (sivu 2)
Oy Firma Ab: Biotekniikan laboratorio
Oy Firma Ab: Osastot
Oy Firma Ab: Kotisivu

VÄLTÄ ORPOJA SIVUJA

Palvelusi kotisivu ei ole aina ainoa sivu, josta käyttäjät saapuvat sivuillesi. Muut WWW-sivujen tekijät saattavat tehdä omille sivuilleen linkkejä suoraan sivujesi sisällä oleviin kohtiin. Myös hakukoneet saattavat antaa käyttäjälle linkin suoraan yksittäiselle palvelusi sisällä sijaitsevalle sivulle.

Tämän vuoksi sinun täytyy pitää huoli siitä, että jokaiselta tekemältäsi sivulta löytyy linkki ulospäin; palvelimesi etusivulle ja kenties myös muille asiaan liittyville sivuille. Jos sivusi ei sisällä linkkejä ulospäin ja käyttäjä saapuu esimerkiksi hakukoneen kautta kyseiselle sivulle, hän ei voi siirtyä eteenpäin tutkimaan muita palvelusi tarjoamia sivuja, vaikka hän olisi kiinnostunutkin aiheesta.



Jokaiselta sivulta ulospäin johtavasta navigoinnista on hyötyä myös silloin, jos käyttäjä ei ole tottunut **Back**-painikkeen käyttöön (tai jos hänen selaimessaan ei

ole helppokäyttöistä **Back**-toimintoa!). Jos **Back**-painike on ainoa tie ulos sivuiltasi, saatat tahtomattasi motittaa käyttäjän.

HYVÄ KÄYTTÖLIITTYMÄ

Hyvä käyttöliittymä takaa sen, että sivuillasi viihdytään ja että niille tullaan uudelleenkin. Hyvän käyttöliittymän ominaisuuksia ovat selkeä ja yhtenäinen ulkoasu ja nopea toiminta. Alla käsitellään näitä.

YHTENÄINEN ULKOASU

Kun teet suuria sivukokonaisuuksia, pyri pitämään kaikilla sivuilla yhtenäinen ilme. Tämän avulla lukijasi pystyvät hahmottamaan palvelimesi. Kun sivuillasi on yhtenäinen omaleimainen ulkoasu, niistä jää myös yhtenäinen mielikuva käyttäjien mieleen ja he muistavat palvelimesi pidempään.

Yhtenäiseen ulkoasuun kuuluvat sekä sivujesi graafinen ilme kuvineen ja värityksineen että myös tekstin ulkoasu ja tyyli. Käytä kaikilla sivuilla samantyylistä kuvitusta ja vältä tekstin ulkoasun muuttamista.

Pienet eroavaisuudet ovat kuitenkin sallittuja ja myös tärkeitä. Voit erotella sivujesi eri kokonaisuudet pienillä tyylimuutoksilla kuten värityksellä tai hieman erilaisella kuvituksella.

NAVIGOINNIN ULKOASU

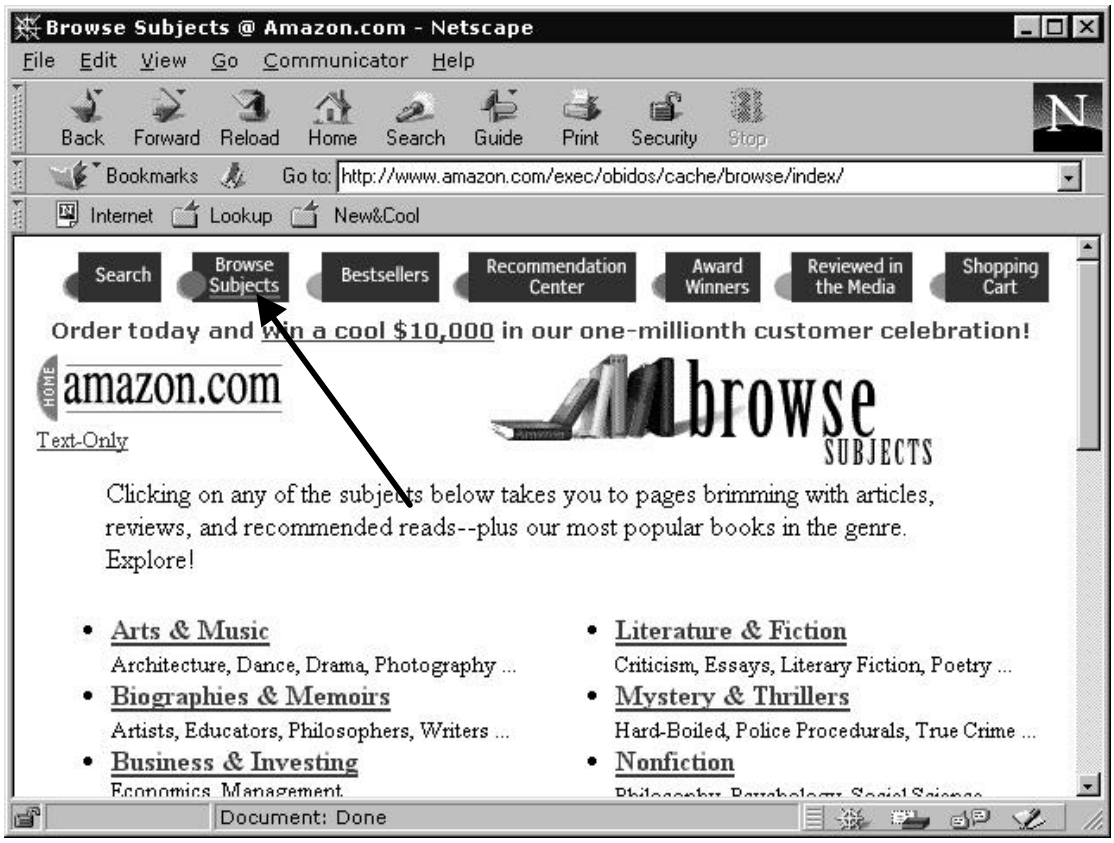
Aiemmissa luvuissa kerrottiin sivujen rakenteen suunnittelusta helposti navigoitaviksi. On myös tärkeää, että käyttäjä pystyy hahmottamaan valitsemasi navigointityylin.

Käyttäjän on pystyttävä aina näkemään mahdollisuutensa siirtyä sivulta toiselle ja hahmottamaan sijaintinsa palvelimessa.

Sivulta toiselle siirtyminen tapahtuu linkkien avulla: käyttäjä voi siirtyä tietyn asiakokonaisuuden sisällä sivulta toiselle tai hän voi vaihtaa asiakokonaisuutta ja siirtyä sivujen kokonaisuudesta toiseen.

Yleinen tapa monilla WWW-sivuilla on tarjota käyttäjälle navigointipalkki, jossa hän näkee palvelussa tarjolla olevat asiakokonaisuudet ja voi siirtyä niiden avulla paikasta toiseen.

Navigointiin kuuluu tärkeänä osana se, miten käyttäjä näkee mahdollisuutensa siirtyä sivulta toiselle. Esimerkiksi Amazon-kirjakaupan (`http://www.amazon.com`) jokaisen sivun ylälaidassa on näkyvissä linkkejä palvelimen eri asiakokonaisuuksiin.



Linkit on toteutettu interaktiivisella kuvalla ja käyttäjällä selaimessa näkyvä asiakokonaisuus on merkitty pallolla ja pienellä alleviivauksella (kuvassa kohta "**Browse Subjects**"). Lisäksi jokaisella sivulla on navigointipalkin alapuolella näkyvissä Amazonin logo, jota napsauttamalla pääsee takaisin palvelun etusivulle.

Navigointilinkit voidaan esittää myös tekstimuodossa. Yleensä kaikista navigointilinkeistä tulee tehdä myös tekstiversio niille käyttäjille, jotka eivät pysty näkemään kuvia. Pelkillä teksteilläkin voidaan tehdä toimivia navigointilinkkejä.

Family.com:in sivuilla navigointi on toteutettu kuvaruudun vasemmassa reunassa olevilla tekstilinkeillä. Jokainen linkki vie eri teemaosan etusivulle ja ne pysyvät näkyvissä palvelun eri sivuilla.

Painikkeet ja linkit voi sijoittaa mihin kohtaan tahansa. Ruudun yläosassa oleva navigointirivi vie aina jonkin verran tilaa ja jättää sivulta näkyviin vähemmän varsinaiselle sisällölle tarkoitettua tilaa. Jos teet navigointirivin ruudun yläosaan, muista tehdä siitä tarpeeksi pienikokoinen.

Toinen tapa on sijoittaa navigointipalkki sivun reunaan. Tähän tarkoitukseen voit käyttää esimerkiksi taulukoita tai kehyksiä (näistä myöhempänä). Esimerkiksi yllä näkyvässä esimerkissä **Family.com**:in linkit ovat yhden taulukon solun sisällä.

Pidä navigointilinkit kaikilla sivuilla saman näköisinä. Lukijasi ei halua varmastikaan opetella jatkuvasti uusia navigointitapoja liikkuessaan sivuillasi. Selkeät linkit ovat nopeita käyttää ja jättävät miellyttävän kuvan palvelimesta.

Muista tehdä lukijallesi jatkuvasti selväksi hänen sijaintinsa palvelimessasi. Navigointipalkki voi esittää korostettuna sen osan, jossa käyttäjä on kullakin hetkellä. Jos linkkisi sisältävät linkkejä eri kokonaisuuksien etusi-

vuille, myös korostettu linkki on silloin toiminnassa ja vie näytöllä olevan kokonaisuuden alkuun.

Alkusivulla oltaessa linkistä ei tapahdu mitään (voit esimerkiksi himmentää painikkeen värejä niin, että käyttäjä huomaa ettei se ole valittavissa).

SELKEYS

Palvelun selkeys on tärkeää, sillä paljon informaatiota sisältävät sivut saattavat hämmentää käyttäjää. Älä sijoita sivuille liikaa linkkejä (poikkeuksena tietysti sivut, jotka sisältävät linkkilistoja tai vastaavia).

Käytä tekstityyppejä ja värejä säästeliäästi - valitse vain muutama tekstityyppi, joita käytät palvelimessasi ja käytä niitä säännönmukaisesti. Sivujen väritys (myös linkkien ja valittujen linkkien värit) kannattaa suunnitella niin ikään huolellisesti.

Kun käytät kaikissa selaimissa toimivia HTML-komentoja mahdollisimman paljon, takaat sen että sivusi toimivat kaikissa selaimissa. Jos käytät selainkohtaisia erikoiskomentoja tai muita tehosteita, on vaarana että sivu ei näykään käyttäjän selaimessa sellaisena kuin tarkoitit.

NOPEUS

Yksi tärkeimmistä seikoista sivujen toiminnassa on niiden nopeus. Mitä nopeammin sivut ilmestyvät ruudulle, sitä enemmän käyttäjät jaksavat selailla palvelimessa olevia tietoja. Suurin osa Internet-käyttäjistä on verkossa modeemin välityksellä ja nekin käyttäjät, joilla on käytössään nopea verkkoyhteys, saavat usein sivut verkon ruuhkaisuudesta johtuen lähes modeemin hitaudella.

Sananlasku "yksi kuva vastaa tuhatta sanaa" pitää paikkansa myös verkossa. HTML-sivut ovat yleensä vain muutaman kilotavun kokoisia, kun taas jo pienikin kuva ylittää kooltaan tekstitiedoston moninkertaisesti. Tästä seuraa se, että mitä enemmän ja suurikokoisempia kuvia sivuillasi on, sitä hitaammin ne latautuvat.

Kuvien pienentäminen on mahdollista, mutta vaatii joskus hieman vaivaa. Kuvien pienentäminen on kuitenkin sen arvoista, sillä nopeammin toimivat sivut ovat suositumpia.

Kuvien tiedostokokoa voi pienentää monella tavalla:

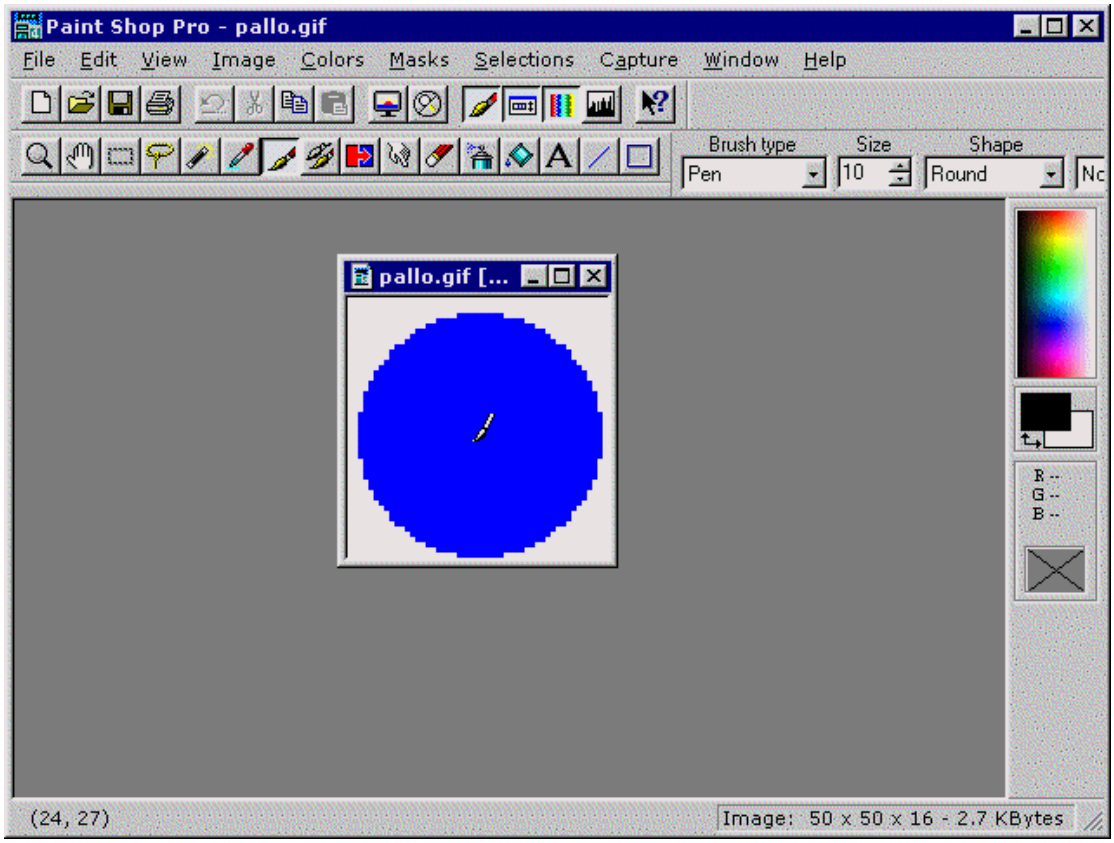
- Pienennä kuvan fyysistä kokoa. Tee heti aluksi kuvista melko pieniä (ota mallia verkossa olevista palvelimista) tai kokeile pienentää valmiita kuvia. Kuvien pienentäminen onnistuu **Paint Shop Prossa Image**-valikon toiminnolla **Resample**. Tämä toiminto pienentää kuvan mahdollisimman hyvälaatuisesti. Jos suunnittelet sivujasi tietokoneella, jolla on hyvin tarkka näyttö, olet saattanut tehdä kuvistasi liian suuria - kokeile katsella sivuasi 640*480 -näyttötilassa tai pienentää selaimen ikkuna sen kokoiseksi.

- Vähennä kuvan värimäärää. Mitä vähemmän kuvassa on bittitasoja, sitä pienikokoisempi se on. Kuvien värimäärän vähentäminen tapahtuu **Paint Shop Prossa Colors**-valikon toiminnolla **Decrease Color Depth**. Kokeile käyttää värien vähentämisen yhteydessä rasterointia - lopputuloksena oleva kuva saattaa näyttää yllättävän hyvältä! Usein kuvissa on useita lähellä toisiaan olevia sävyjä samasta väristä - niiden yhdistäminen ei heikennä kuvan laatua paljoakaan mutta saattaa vähentää värimäärää niin paljon, että kuva voidaan tallentaa vähemmillä bittitasoilla.

- Tallenna kuva **jpeg**-muodossa. Tässä muodossa tallennetut kuvat ovat usein paljon pienempiä kuin vastaavat **gif**-kuvat. Koska **jpeg** hukkaa kuvien yksityiskohtia tallennettaessa, se ei kuitenkaan sovi kovin hyvin piirroksille, joissa on vähän värejä. Parhaat tulokset saadaan pakattaessa valokuvista skannattua materiaalia. Kokeile myös erilaisia **jpeg**in pakkausvahvuuksia ja valitse sellainen vahvuus, jossa kuvan näyttää vielä hyvältä.

Voit myös nopeuttaa vasteaikoja tallentamalla **gif**-kuvat **interlace**-muodossa ja **jpeg**-kuvat **progressive**-muodossa. Kuvat eivät lataudu silloin yhtään nopeammin, mutta käyttäjä saa ruudulleen nopeammin vastetta palvelimelta.

Sivun latausta nopeuttaa myös, jos määrittelet kuvakomennossa kuvan ulottuvuudet. Tämä tapahtuu lisäämällä kuvakomentoon attribuutit **`width`** ja **`height`**. Saat selville kuvan ulottuvuudet **Paint Shop Prossa** viemällä hiiren osoittimen kuvan päälle. Kuvan ulottuvuudet näkyvät ikkunan oikeassa alanurkassa (samassa paikassa näet myös kuvan koon pakkaamattomana):



Kun lisäät sivulle kuvakomennon, lisää siihen kuvan korkeus ja leveys. Esimerkissä pallon leveys ja korkeus olivat 50 pikseliä:

```
<img src="pallo.gif" width=50 height=50>
```

Kun olet määritellyt kuvan koon, selain pystyy näyttämään sivun tekstisisällön paljon nopeammin. Normaalisti selain joutuu lataamaan hieman kuvan alkua ennen kuin se tietää, minkä kokoinen kuva tulee olemaan ruudulla. Ennen tätä selain ei pysty näyttämään kuvan ympärille ja sen jälkeen tulevia tekstejä.

Kun olet lisännyt kuvakomentoon leveys- ja korkeusattribuutit, selain pystyy varaamaan kuvalle sopivankokoisen tilan jo ennen kuin se aloittaa varsinaisen kuvan latauksen. Tällä tavalla käyttäjä saa sivun tekstisisällön nopeasti ruudulleen ja kuvat latautuvat niille varatuille paikoille normaalilla nopeudella.

Kun selain on ladannut kuvan sivulle, kuva jää useimmiten selaimen välimuistiin ainakin joksikin aikaa. Voit käyttää tätä hyväksesi käyttämällä samaa kuvaa sivulla uudelleen. Jos samaa kuvaa käytetään sivulla useita

kertoja, se ladataan vain kerran. Toisella kerralla kuva tulee näkyviin selaimen välimuistista.

Voit käyttää samasta kuvasta myös eri kokoisia ja -muotoisia versioita käyttämällä hyväksesi leveys- ja korkeusattribuutteja. Jos annat näihin attribuutteihin jonkin muun arvon kuin kuvan oikean koon, selain suurentaa tai kutistaa kuvaa vastaavasti. Tällä tavalla voit tehdä sivuillesi esimerkiksi isoja ja pieniä palloja:

```
<img src="pallo.gif" width=100 height=100>
<img src="pallo.gif" width=25 height=25>
```

Voit myös venyttää kuvaa enemmän vaaka- tai pystysuunnassa:

```
<img src="pallo.gif" width=100 height=50>
```

Jos kuvien pienentämisestä ja tekstimäärän vähentämisestä huolimatta sivusi latautuu vieläkin hitaasti, paloittele sivusi useaksi eri sivuksi. Tällä tavalla käyttäjä saa nopeasti vastinetta napsautuksilleen ja voi siirtyä yhden sivun luettuaan seuraavalle sivulle.

Jos teet useita peräkkäisiä sivuja, voit ladata seuraavan sivun kuvia sisään jo samalla kun käyttäjä lukee edellistä sivua käyttämällä hyväksesi leveys- ja korkeusattribuutteja.

Oletetaan, että teet palvelua, jossa on kaksi sivua, **`sivu1.html`** ja **`sivu2.html`**. Suurin osa käyttäjistä tulee liikkumaan sivuilla siten, että he lukevat ensin sivun 1 ja siirtyvät sen jälkeen sivulle 2. Sivu 2 sisältää pari suurikokoista kuvaa, jotka haluaisit ladata sisään samalla kun käyttäjä lukee ensimmäistä sivua. Kuvien nimet ovat esimerkiksi **`suurikuva1.gif`** ja **`suurikuva2.gif`**.

Voit lisätä sivun 1 HTML-tiedostoon kuvakomennot, jotka lataavat nämä kuvat sisään. Koska niitä ei ole kuitenkaan tarkoitettu käytettäväksi sivulla 1, voit "piilottaa" ne muuttamalla ne yhden pikselin kokoisiksi **`width`**- ja **`height`**-attribuuteilla.

Esimerkki tiedostosta **`sivu1.html`**:

```
<p>Tässä on pitkä artikkeli jota käyttäjä lukee jonkin
aikaa.
<p><a href="sivu2.html">Seuraavalle sivulle</a>
<p>
<!-- tässä on tiedoston loppu jonne -->
<!-- sijoitetaan kätketyt kuvat -->
<img src="suurikuva1.gif" width=1 height=1>
<img src="suurikuva2.gif" width=1 height=1>
```

Koska kuvakomennot ovat tiedoston lopussa, ne eivät vaikuta sivun alkuosan latautumiseen millään lailla. Samalla kun käyttäjä lukee sivua, kuvat latautuvat sivulle (ja tulevat näkyviin vain yhden pikselin kokoisina, käytännössä siis miltei näkymättöminä). Kun käyttäjä siirtyy seuraavalle sivulle, kuvat löytyvät selaimen välimuistista ja tulevat näkyviin välittömästi. Jos käyttäjä siirtyy seuraavalle sivulle ennen kuin kuvat ovat latautuneet, ne ladataan kyseisellä sivulla normaalisti.

VASTEAJAT

Sivusi vasteaika ja latausaika vaikuttavat sen käyttömukavuuteen. Jo kymmenen sekunnin odotusaika katkaisee käyttäjän mielenkiinnon ja hänen ajatuksensa alkaa harhailla muihin asioihin[4]. Silloin käyttäjä saattaa muistaa jonkin toisen mielenkiintoisen WWW-sivun ja siirtyä sinne.

Rakenna siis sivusi siten, että niiden vasteaika on mahdollisimman nopea. Käytä kuvissa leveys- ja korkeusattribuutteja ja lisää kuviin selitystekstit (tällä tavalla käyttäjä näkee kuvien merkityksen jo ennen kuin kuvia on ehditty ladata).

YLLÄPITO

Kun sivusi ovat valmiit ja julkistettu, niiden ylläpito ei lopu vielä siihen. Sivuja täytyy pitää aktiivisesti yllä, niin että ne toimivat jatkuvasti eivätkä tuota käyttäjälle pettymyksiä.

---

[4] **http://www.useit.com/papers/responsetime.html**
http://www.lava.net/%7edewilson/web/rant.html

DEBUGGAUS

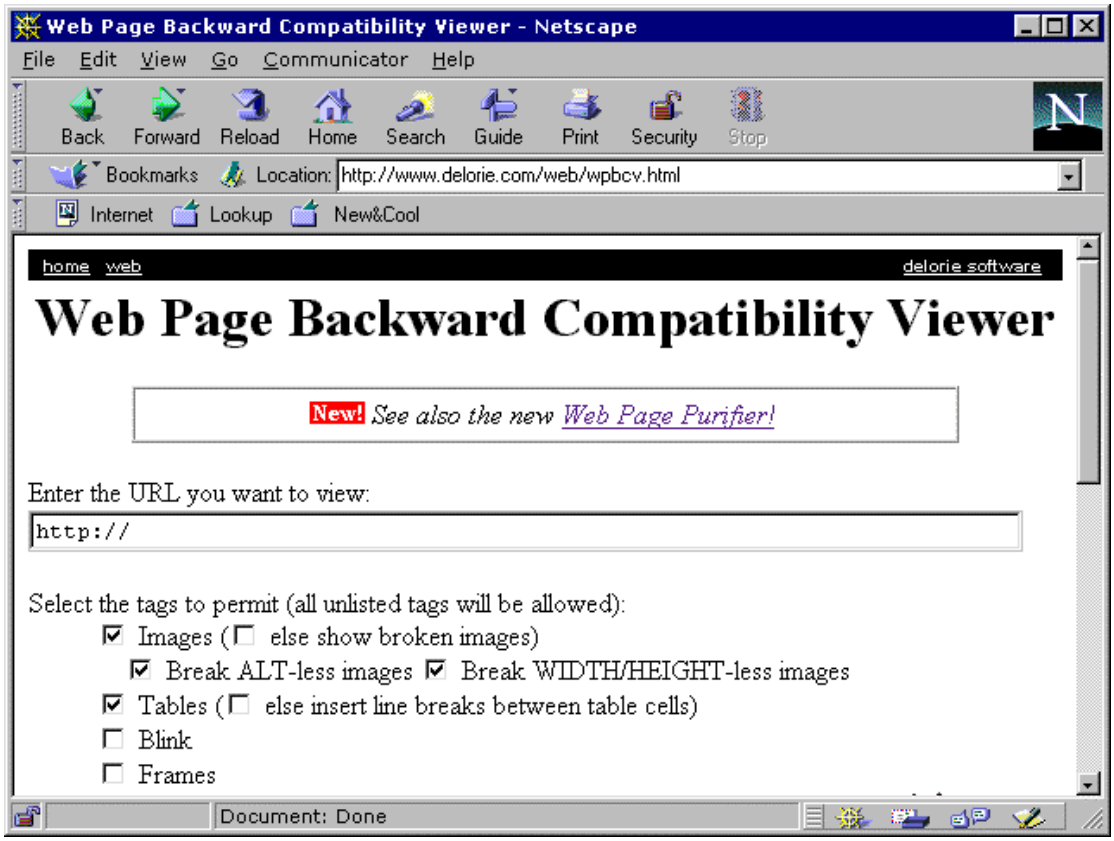
Ennen sivujen julkistusta sivut tulee testata ja katsoa, että ne toimivat halutulla tavalla.

Kokeile aluksi selailla kaikki sivut läpi. Älä käytä selaimen **Back**-painiketta vaan kulje sivuillasi käyttäen omia navigointikeinojasi. Pääsetkö kaikille sivuille ja pääsetkö sivuilta myös eteenpäin?

Kytke seuraavaksi kuvien lataus pois päältä selaimestasi. Tyhjennä selaimen välimuistit, niin että selaimesi ei näytä yhtään sivuillasi olevaa kuvaa (Netscapessa välimuistit tyhjennetään kohdasta **Edit/Preferences/Advanced/Cache**). Katsele sivusi uudelleen läpi. Saatko selvää kaikista sivuista? Pystytkö navigoimaan?

Kokeile katsella sivujasi myös muilla selaimilla kuin normaalisti käyttämälläsi. Hanki vanhempia versioita selaimista ja katso, toimivatko sivusi niillä. Voit käyttää myös palvelua nimeltään "**Web Page Backward Compatibility Viewer**" osoitteessa

`http://www.delorie.com/web/wpbcv.html`



Sen avulla voit katsoa sivujasi ilman kuvia, taulukoita jne. Anna testattavan sivun URL tekstilaatikkoon sivun keskellä ja poista rastit kaikista niistä HTML-ominaisuuksista, joita et halua käytettävän. Kokeile, miten sivusi toimivat ilman näitä.

PIDÄ SIVUT AJAN TASALLA

Muista pitää sivusi ajan tasalla. Jos sivusi näyttävät muuttumattomilta pitkään ja etenkin, jos ne sisältävät vanhentunutta tietoa, käyttäjät eivät mielellään katsele sivujasi.

Pidä sivuillasi oleva tieto ajan tasalla ja muuta sivujen ulkonäköä säännöllisesti niin, että sivuillesi uudelleen saapuvat käyttäjät huomaavat sivuilla lukemisen arvoisia muutoksia.

Kerro näkyvällä paikalla sivuillasi tapahtuneista muutoksista. Tällä tavalla vierailijat, jotka käyvät silloin tällöin tarkistamassa onko sivuillasi uutta luettavaa, huomaavat syvemmän tarkastelun olevan tarpeen.

Älä kuitenkaan muuta turhaan palvelusi ulkoasua tai rakennetta. Säännölliset lukijasi ovat tottuneet käyttämään navigointitoimintojasi ja rakennettasi. Jos palvelusi muuttuu usein toisenlaiseksi, saatat menettää uskollisia lukijoita.

HAJAUTA TYÖ

Kun sivukokonaisuutesi laajenee, sen tietojen ja rakenteen ylläpitäminen saattaa muuttua ylivoimaiseksi. Harkitse silloin työn hajauttamista: etsi sivujesi eri aihepiireille ylläpitäjät, jotka pitävät yllä kutakin aihepiiriä.

Kun työ on jaettu usean päivittäjän kesken, se ei muodostu kenellekään ylivoimaiseksi taakaksi. Tällä tavalla sivusi pysyvät hyvin ylläpidettyinä.

Jos ylläpitäjät päivittävät sivuja palvelinkoneelle omalla tunnuksellaan, varmista että jokaisella ylläpitäjällä on oikeus sijoittaa tiedostoja palvelimen hakemistoihin. Ota tarvittaessa yhteyttä operaattoriisi.

Muista koordinoida ylläpito, jotta eri ylläpitäjät eivät muokkaa yhtäaikaa samoja tiedostoja. Tällaisessa tilanteessa toisen ylläpitäjän tekemät muutokset saattavat hävitä.

Muista sopia ylläpitäjiesi kanssa palvelimesi ulkoasun, rakenteen ja navigoinnin toteuttamisesta. Silloin ylläpitäjät osaavat rakentaa uusia sivuja palvelimen yhteisen linjan mukaisesti.

TARKISTA LINKIT

Verkkosivut elävät jatkuvasti: palvelimien rakennetta järjestellään uusiksi, osoitteet vaihtuvat ja joskus jopa samassa osoitteessa sijaitseva palvelin muuttuu toisenlaiseksi!

Onkin tärkeää, että tarkistat säännöllisesti sivuillasi olevat linkit. Toimimattomat linkit paljastavat nopeasti lukijoillesi, että et ole kiinnittänyt huomiota sivujesi päivitykseen. Jos linkit ovat muuttuneet, korjaa ne välittömästi omiin HTML-tiedostoihisi.

Jos muutat itse sivujesi rakennetta, ota huomioon muualta sivuillesi linkkejä tehneet. Ystävällinen tapa onkin jättää vanhaan osoitteeseen tiedotus siitä, että sivuilla on uusi osoite ja myös tarjota linkki tähän osoitteeseen. Kun käyttäjä seuraa vanhentunutta linkkiä tällaiselle sivulle, hän voi siirtyä välittömästi oikeaan paikkaan.

Voit myös tehdä sivulle toiminnon, joka siirtää käyttäjän automaattisesti oikealle sivulle. Lue tästä aiempaa, kohdasta jossa kerrottiin etusivun tekemisestä.

LINKKIENTARKISTUSOHJELMIA

Verkosta löytyy palveluita ja ohjelmia, jotka tarkistavat puolestasi sivuillasi olevat linkit. Monet edistyneemmät HTML-editorit osaavat niin ikään tarkistaa sivuilla olevia linkkejä. Tällaisilla ohjelmilla sivujen tarkistaminen käy näppärästi.

Linkkientarkistusohjelmia löydät **Tucows:ista** osoitteesta

http://www.eunet.fi/tucows/htmlval95.html

OTA VASTAAN PALAUTETTA

Jos sivuillasi on virheitä tai vanhentunutta tietoa, jotkut lukijoistasi saattavat olla avuliaita ja kertoa sinulle siitä. Lukijat saattavat myös lähettää sinulle parannusehdotuksia tai toivomuksia sivujen rakenteesta.

Tämä on kuitenkin mahdollista vain, jos tarjoat sivuillasi mahdollisuuden lähettää palautetta. Yksinkertaisimmillaan palautemahdollisuus on sivuilla mainittu sähköpostiosoite. Voit myös tehdä linkin, jota napsauttamalla käyttäjän selain käynnistää sähköpostiohjelman ja aloittaa uuden viestin määriteltyyn osoitteeseen. Tämä tehdään URL-osoitteella, jonka alussa on teksti **`mailto`**:

```
<a href="mailto:osoite@palvelin.fi">Lähetä
palautetta!</a>
```

Voit myös rakentaa sivuillesi palautelomakkeen, jonka täyttämällä käyttäjäsi voivat lähettää sinulle palautetta.

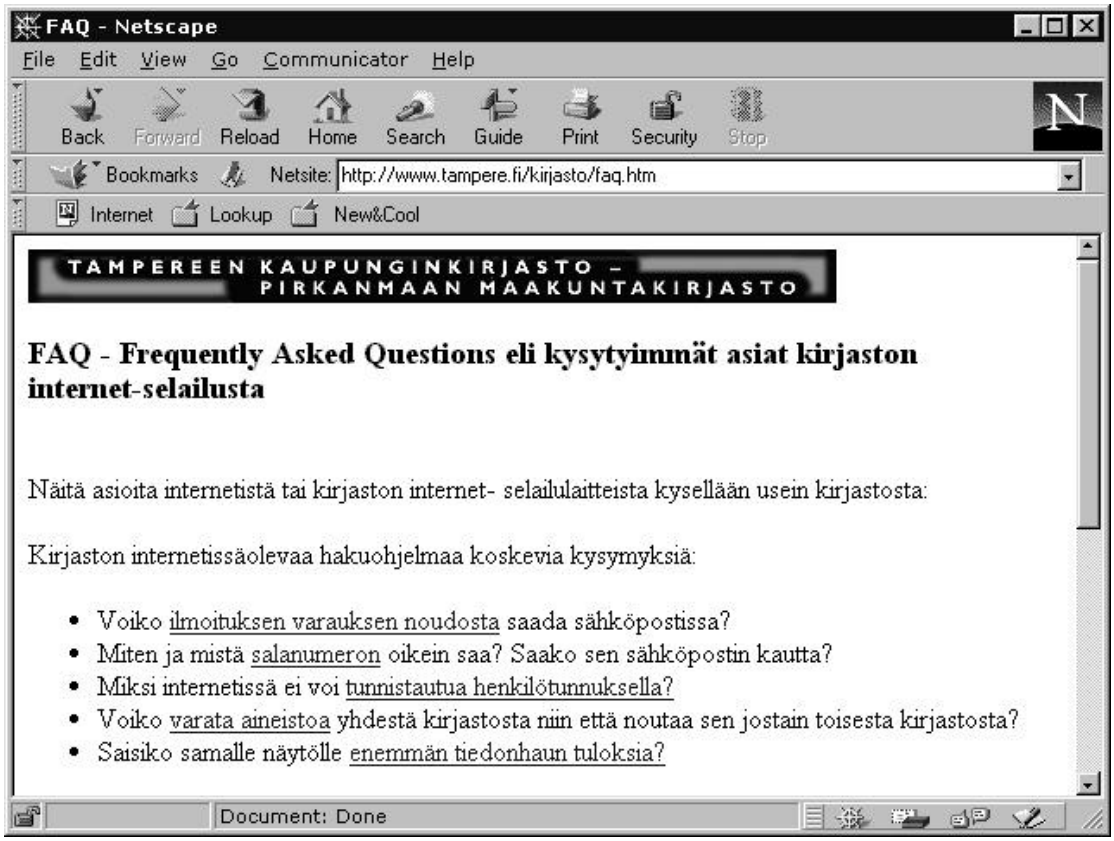
Muista lukea saamasi palaute ja reagoida siihen. Palautemahdollisuus, johon kukaan ei koskaan reagoi, antaa huonon kuvan palvelustasi.

Jos suunnittelet kokonaista WWW-palvelinta, yleinen tapa on tehdä sähköpostiosoite **`webmaster@palvelin.fi`**, jossa otetaan vastaan palautetta palvelimesta ja sen toiminnasta. Voit neuvotella operaattorisi kanssa tällaisen osoitteen tekemisestä ja siihen tulevan postin ohjaamisesta edelleen sinulle tai ylläpitoryhmällesi.

TEE FAQ

Jos sivusi tulevat suosituiksi ja tarjoat palautemahdollisuuden, tulet saamaan paljon palautetta sivuistasi. Useimmiten käyttäjät kysyvät asioita, joihin palvelimestasi ei löydy vastausta. Tällaisia kysymyksiä saattaa tulla paljonkin.

Näihin kysymyksiin voi vastata jo ennen kuin ne esitetään. Voit tehdä palvelimeesi **FAQ**-sivut. Termi **FAQ** tulee sanoista **Frequently Asked Questions**, eli Usein kysytyt kysymykset. Internetissä yleinen tapa on tehdä sivuja, joissa luetellaan usein kysyttyjä kysymyksiä ja annetaan niihin vastaukset.



Kun teet palvelimeesi **FAQ**-sivut, vaikutat saamasi palautteen määrään positiivisesti - et saa enää samoja kysymyksiä uudelleen ja uudelleen (ja vaikka saatkin, voit ohjata kysyjän lukemaan **FAQ**-sivua). Sen sijaan voit keskittyä jäljelle jäävän palautteen käsittelemiseen (kommenttien, lisäysehdotuksien ja sellaisten kysymyksien, joihin et ole vielä ehtinyt tehdä **FAQ**-kohtaa).

FAQ-sivuista on hyötyä myös muille palvelimesi käyttäjille - niitä selaamalla lukijat oppivat uusia asioita jo ennen kuin he ovat edes ehtineet ajatella niitä.

KÄYTÖN SEURANTA

Valmiiden sivujen käyttötilastot ovat mielenkiintoista luettavaa. Niiden avulla näet, mitä sivujasi luetaan eniten, mistä lukijasi tulevat, mitkä linkit johtavat sivuillesi ja mitä selaimia lukijasi käyttävät.

LOKIT

WWW-palvelinohjelmistot keräävät sivujen käyttötietoja lokitiedostoihin. Nämä tiedostot sijaitsevat palvelinkoneella ja kasvavat jatkuvasti. Sitä mu-

kaa kuin palvelinkoneelle tulee sivunlukupyyntöjä, palvelin lisää tiedon näistä pyynnöistä lokitiedoston loppuun.

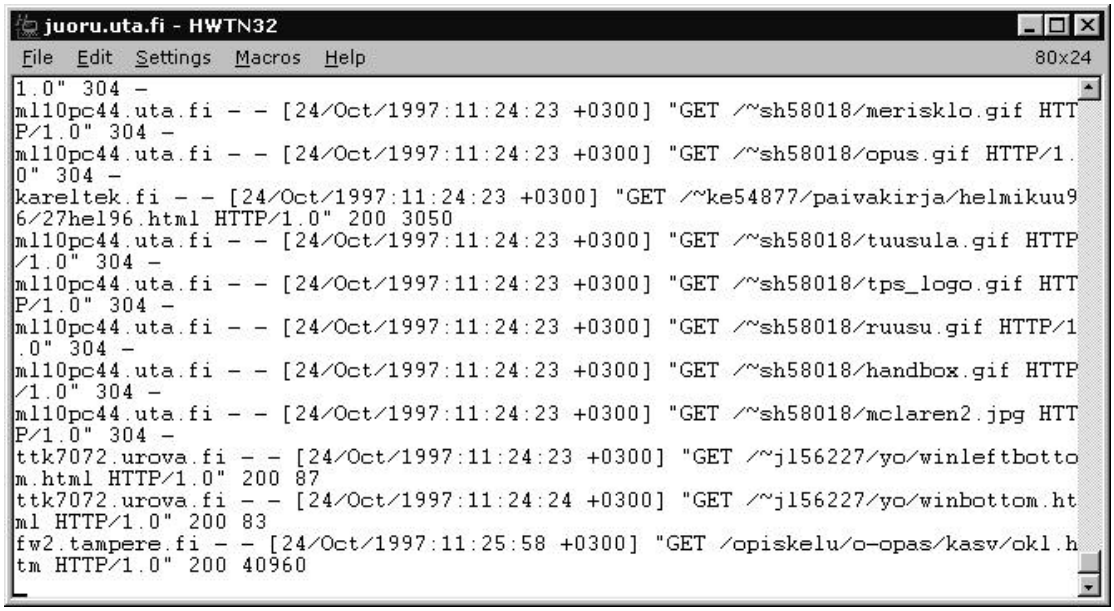
Jos Internet-operaattorisi on tarjonnut mahdollisuuden tutkia palvelimen lokitiedostoja, voit myös itse katsella niitä. Tiedon siitä, missä hakemistossa ja minkä nimisiä oman palvelimesi lokitiedostot ovat, saat operaattorisi käyttöohjeista tai asiakaspalvelusta.

Lokitiedostoja on yleensä neljä kappaletta: tiedostoon nimeltään **`access`** kerätään tietoja luetuista tiedostoista, tiedostoon **`errors`** palvelimen antamia virheilmoituksia (esimerkiksi puuttuvat sivut), tiedostoon **`agent`** tietoja käyttäjien selainmerkeistä ja tiedostoon **`referrer`** tietoja siitä, miltä sivulta palvelimen kullekin sivulle on saavuttu.

Jos palvelinkoneessa on Unix-käyttöjärjestelmä ja sinulla on lukuoikeudet lokitiedostoja sisältämään hakemistoon, voit katsella palvelimen käyttölokia reaaliaikaisena komennolla

```
tail -f access
```

Voit lopettaa listauksen painamalla näppäimistöltäsi **Ctrl-C**. Listaus näyttää tältä:



Lokin jokaisen rivin alussa näet lukijan käyttämän tietokoneen nimen ja rivin loppupäässä hänen hakemansa tiedoston osoitteen (ilman **`http://koneennimi`** -alkuosaa). Muita lokiin kerättäviä tietoja ovat haun

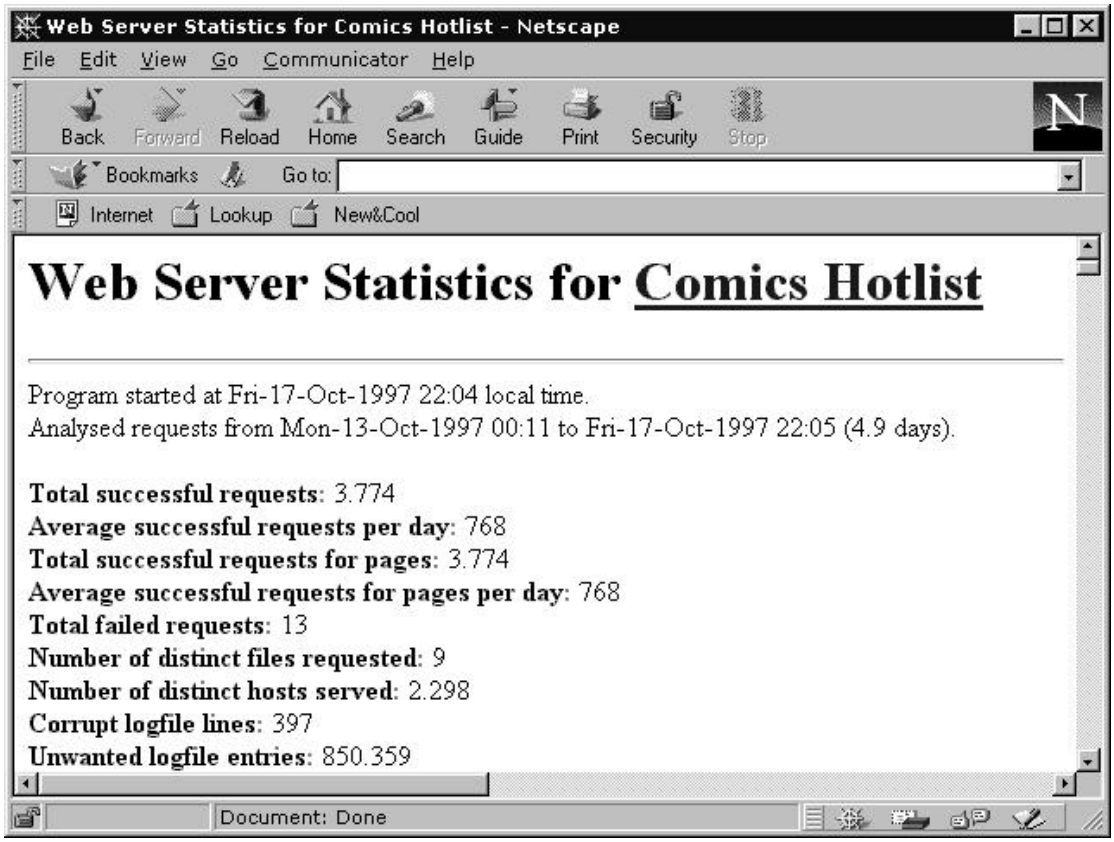
päivämäärä ja kellonaika, hakutuloksen koodi (**200** = haku sujui onnistuneesti) ja tiedoston koko.

Kun palvelimesta luetaan sivu, joka sisältää kuvia, jokaisesta kuvasta tulee yksi hakupyyntö lokitiedostoon.

LOKIEN SEURANTATYÖKALUJA

Lokitiedostot ovat suurikokoisia, joten niistä on vaikea saada lukemalla esiin mitään tietoja sivun käytöstä. Tämän vuoksi lokitiedostojen analysointiin on kehitetty monia työkaluohjelmia, joiden avulla tiedostoista voidaan ottaa irti monenlaisia tietoja.

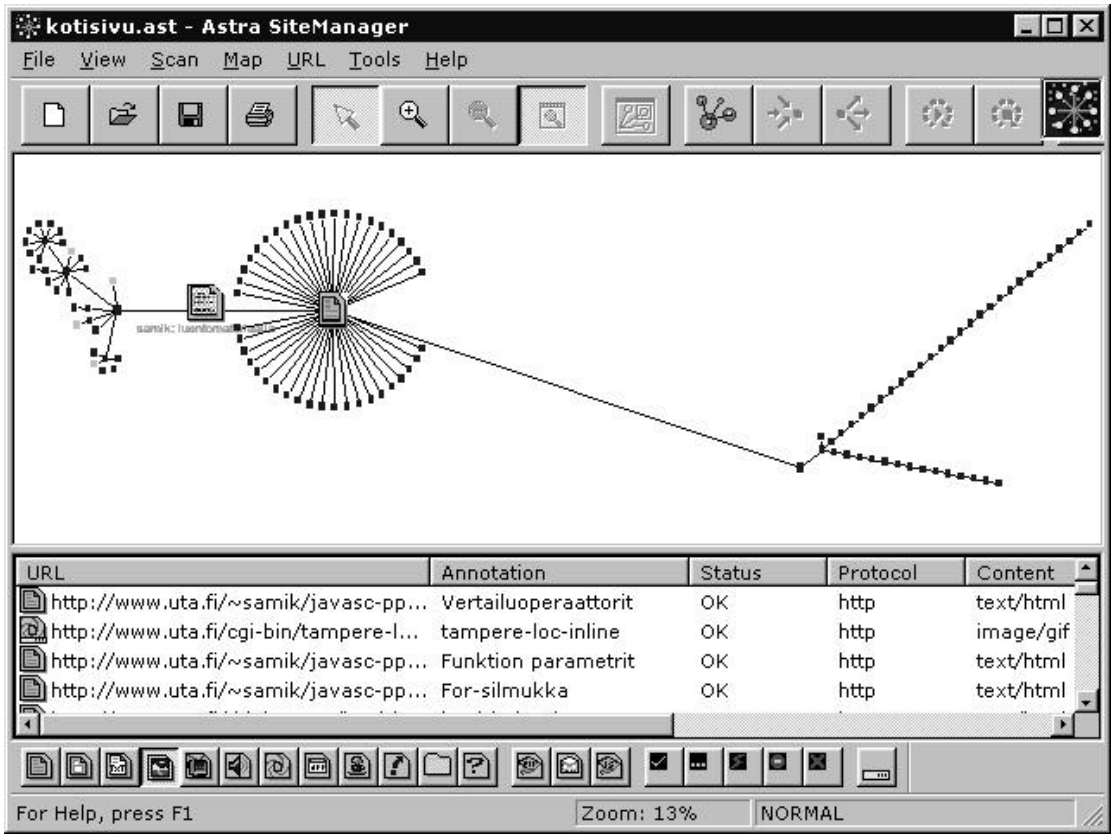
Yksi yleisimpiä lokien analysointiohjelmia on palvelinkoneella ajettava **analog**. **Analog** tekee lokitiedostoista raportin, josta näet sivujen käytön, lukijoiden kotimaan ja monia muita tietoja.



Löydät **Analogin** kotisivun osoitteesta:

`http://ftp.sunet.se/pub/www/utilities/analog`

Windows-ohjelmalla nimeltään **Astra Sitemanager** voit tehdä sivuistasi kartan ja tutkia myös sitä, miten lukijat ovat liikkuneet sivuillasi ja mitkä reitit ovat suosituimpia sivujesi sisällä. Koska **Astra** on Windows-ohjelma, voit kopioida palvelimen lokitiedostot omalle koneellesi ja käsitellä niitä siellä. Jos WWW-palvelin pyörii Windows-koneessa, voit ajaa **Astraa** myös suoraan palvelinkoneessa.



Voit kysyä omalta Internet-operaattoriltasi, onko käyttämäsi palvelimen lokeista saatavilla analysoituja tietoja ja onko lokeista valmistettu valmiita raportteja.

Windows-pohjaisia lokienanalysointiohjelmia löydät ohjelma-arkistoista, esimerkiksi **Tucows:ista** sivulta

`http://www.eunet.fi/tucows/log95.html`

INTRANET-SUUNNITTELU

Intranet on termi, joka tarkoittaa sisäiseen käyttöön suunniteltuja WWW-sivuja. Organisaatiolla saattaa olla sivuillaan tarjolla tietoa, joka on tarkoitettu vain omalle henkilökunnalle. Tällaiset sivut voidaan sijoittaa ulko-

maailmalta piiloon joko omaan palvelimeensa tai WWW-sivujen erilliseen haaraan. Yleensä sivut suojataan joko salasanalla tai niille sallitaan pääsy vain tietyistä osoitteista (voit kysyä sivujen salasanasuojaamisesta lisäohjeita operaattoriltasi).

Koska intranet-sivuja luetaan vain organisaatiosi sisältä, voit jättää sivuiltasi pois mainos- ja imagonluontiin tarkoitetut kuvat. Voit myös suunnitella sivujen rakenteen olettaen, että lukijat tietävät tarkemmin, minne he ovat menossa ja mitä he haluavat löytää.

Intranet-sivujen etusivu voi sisältää paljon linkkejä, jotka osoittavat suoraan tärkeimpiin paikkoihin sivujesi sisällä sekä myös tärkeimpiin organisaatiosi käyttämiin linkkeihin Internetissä.

Extranet on intranetin ulkomaailmalle suunnattu vastine. Sillä toteutetaan esimerkiksi sivut, jotka on tarkoitettu nähtäväksi vain yrityksen asiakkaille. **Extranet**-sivut ovat niin ikään suojattuja, yleensä salasanaa apuna käyttäen.

KUN SIVUT OVAT VALMIITA - JULKISTAMINEN

Kun sivut on asennettu palvelimeen ja testattu, että kaikki osat toimivat suunnitelmien mukaan, on aika kertoa maailmalle sivujen olemassaolosta ja hankkia niille lukijoita.

Jos olet tehnyt kotisivut, voit pyytää palvelimesi ylläpitäjää (operaattoriasi) lisäämään kotisivusi palvelimessa olevaan kotisivujen listaan (joissakin koneissa tämä tapahtuu automaattisesti). Tämän jälkeen palvelimesi tietoja lukevat vierailijat saattavat huomata uuden sivun ja tulla tutkimaan sitä.

Jos haluat enemmän lukijoita, voit harrastaa myös aktiivista mainostusta. Maailmalla pidetään yllä useita uutuuslistoja, joihin kerätään säännöllisesti uusia WWW:stä löytyviä palveluita. Näitä listoja lukevat käyttäjät, jotka haluavat nähdä, mitä uutta verkko tarjoaa heille tai jotka tutustuvat verkon tarjontaan. Monille uutuuslistoille voi lähettää vapaasti tietoa uusista WWW-sivuista.

SUOMALAISIA UUTUUSLISTOJA

Suomessa on useita palveluita, jotka keräävät tietoa uusista WWW-sivuista. Voit ilmoittaa sivusi näihin palveluihin ilmaiseksi.

Alku: `http://alku.hrsk.edu.fi/`

Cardinal Company Index: `http://www.cardinal.fi/index/`

Linka: `http://www.pcuf.fi/linka/`

Lisko: `http://www.lisko.oulu.fi/`

Serveri: `http://www.serveri.net/`

Webstudio: `http://www.WebStudio.fi/`

ULKOMAISIA LINKKILISTOJA

Ulkomailta löydät valtavan määrän linkkilistoja. Jotkut ovat erikoistuneet tiettyihin aihepiireihin, toiset taas ovat yleisluonteisempia hakemistoja. Ehkäpä laajin näistä hakemistoista on **Yahoo**, jonne sivusi kannattaakin ilmoittaa ensimmäisenä.

http://www.yahoo.com/

Yahoo sisältää hakemiston verkossa sijaitsevista ilmoituspalveluista. Täältä löydät sivuja, joiden avulla voit lisätä osoitteesi lukuisille verkosta löytyville listoille ja ohjeita siitä, minne sivuistasi kannattaa ilmoittaa. Jotkut ilmoituspalvelut ovat kaupallisia. Listan ilmoituspalveluista löydät osoitteesta

**http://www.yahoo.com/Computers_and_Internet/
Internet/World_Wide_Web/Announcement_Services/**

Yahoosta löydät niin ikään listan verkosta löytyvistä linkkihakemistoista. Tämä lista löytyy osoitteesta

**http://www.yahoo.com/Computers_and_Internet/
Internet/World_Wide_Web/Searching_the_Web/
Web_Directories/**

Tutustu myös sivuun

http://ep.com/faq/webannounce.html

Tältä sivulta löydät lisää vinkkejä paikoista, joihin voit ilmoittaa vastavalmistuneet sivusi.

HAKUKONEET

Voit ilmoittaa sivusi myös WWW-hakukoneisiin. Jos sivuillesi on yksikin linkki muilta WWW-sivuilta, sivusi löytävät tiensä hakukoneiden listoille ennemmin tai myöhemmin. Voit kuitenkin nopeuttaa sivujesi löytymistä hakukoneista ilmoittamalla ne suosituimpiin hakupalveluihin. Hakupalveluita ovat

Alta Vista: **http://www.altavista.digital.com/**

Excite: **http://www.excite.com/**

Hotbot: **http://www.hotbot.com/**

Infoseek: **http://www.infoseek.com/**

Lycos: **http://www.lycos.com/**

Löydät ohjeet linkin lisäämisestä hakupalveluiden kotisivuilta.

UUTISRYHMÄT

news:comp.infosystems.www.announce
news:sfnet.viestinta.www.uutuudet

Uutisryhmä **comp.infosystems.www.announce** sisältää tiedotuksia uusista palveluista Internetissä. Voit postittaa tänne tiedon omista WWW-sivuista. Ilmoituksesi saa nopeasti lukijoita ympäri maailmaa, ja voitkin odottaa ensimmäisiä vieraita jo muutaman tunnin sisällä ilmoituksesi julkaisemisesta. Tätä uutisryhmää toimitetaan (artikkelit eivät siis tule automaattisesti näkyviin), joten lue lähettämisohjeet ensin osoitteesta

http://vader.boutell.com/%7Egrant/charter.html

Uutisryhmä **sfnet.viestinta.www.uutuudet** on suomalainen vastine **comp.infosystems.www.announce** –ryhmälle. Tänne voit postittaa tiedot uusista sivuistasi, etenkin jos arvelet niiden kiinnostavan erityisen paljon suomalaista yleisöä. Huomaa, että ä-kirjain kirjoitetaan a:na uutisryhmän nimessä!

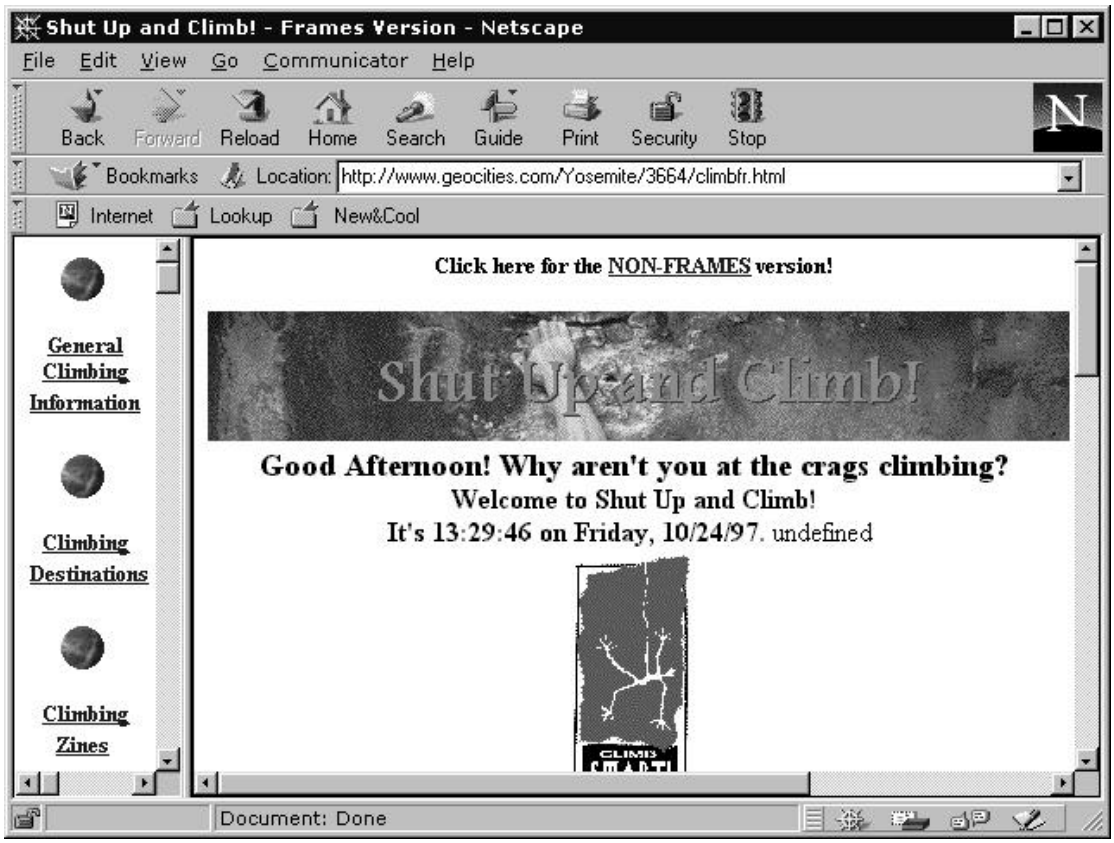
EDISTYNYT SIVUJENSUUNNITTELU

KEHYKSET

Kehykset (**Frames**) muistuttavat hieman HTML-taulukoita. Siinä missä taulukoilla voidaan tehdä sivun sisään osia, jotka on muotoiltu riveiksi ja sarakkeiksi, kehyksien avulla voidaan jakaa itse selaimen kuvaruutu osiin.

Kehyksien avulla voit tehdä esimerkiksi sivun, jossa selaimen ikkunassa yksi reuna on varattu navigointipalkkia varten, yläreunassa on tilaa otsikolle ja loppuosa ikkunasta on varattu varsinaiselle sivulle. Kun lukija katselee sivua, navigointipalkki ja otsikko pysyvät paikoillaan, vaikka lukija rullaisikin sivua alas- ja ylöspäin.

Kuvan esimerkissä ruutu on jaettu kahteen kehykseen: vasemmalla on navigointikehys ja oikealla sivun varsinainen sisältö.



KEHYKSIEN MÄÄRITTELY

Kun sivu määritellään käyttämään kehyksiä, sen alkuun lisätään yksi tai useampia **`frameset`**-komentoja. Tällaisella sivulla ei käytetä ollenkaan **`bo-`**

dy-komentoa. Sivulla annetaan lisäksi komento **noframes**, jonka aloitus- ja lopetuskomennon väliin sijoitetaan tekstiä näytettäväksi niillä selaimilla, jotka eivät osaa näyttää kehyksiä.

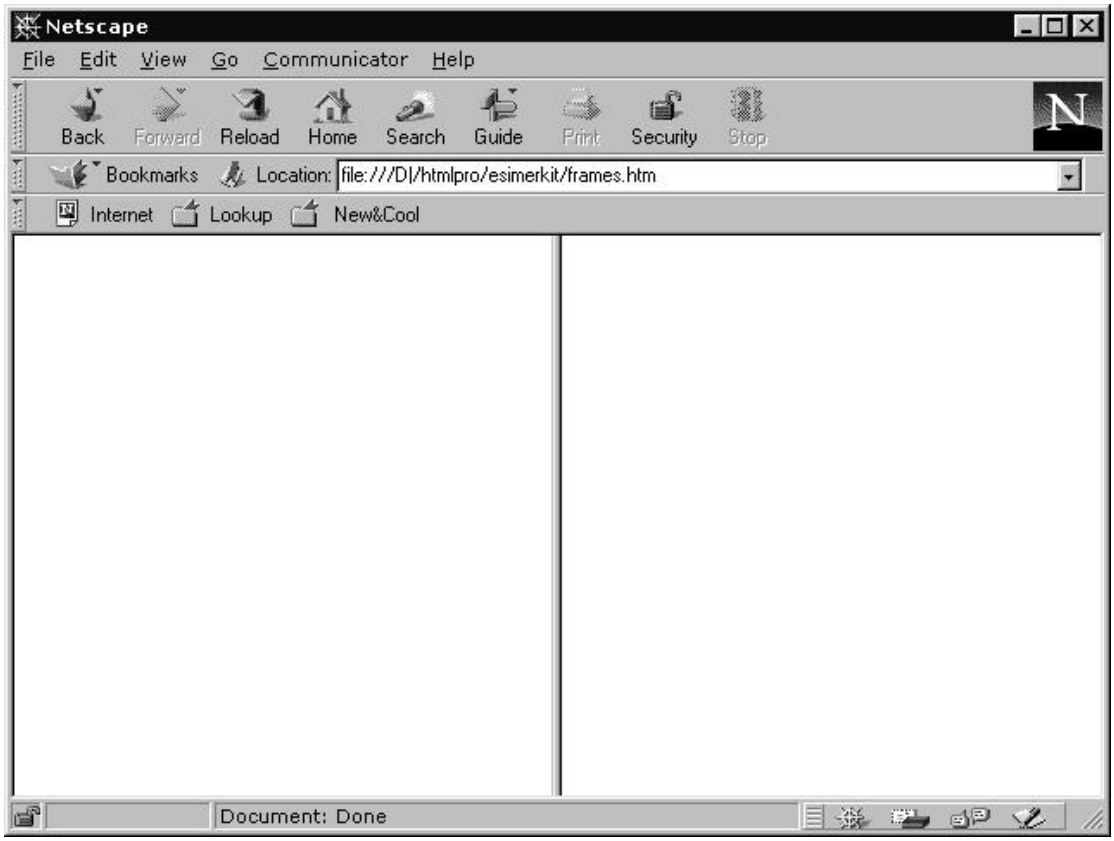
Kehyksiä käyttävän sivun rakenne näyttää seuraavanlaiselta:

```
<html>
<head>
Normaaleja otsikkokomentoja, esimerkiksi <title>
</head>

<frameset>
Kehysmäärittelyt.
</frameset>

<noframes>
HTML-komentoja selaimille, jotka eivät osaa näyttää kehyksiä.
</noframes>
</html>
```

Frameset-komento kertoo, miten näyttö tulee jakaa eri kehyksien kesken. Komennolle voi antaa attribuutit **rows** tai **cols**. Kummallekin attribuutille voi antaa kehyksen koon kolmella eri tavalla: pikselien lukumääränä, prosenttiarvona tai suhteellisena lukuna. Jos haluat jakaa selaimen ikkunan vaakasuunnassa kahteen kehykseen, joista molemmat vievät puolet ikkunan leveydestä, voit käyttää seuraavanlaista **frameset**-komentoa:

```
<frameset cols="50%,50%">
```

Sivu jakaantuu ruudulla kahteen yhtä suureen osaan:



Pelkkä **frameset**-komento ei kuitenkaan yksinään riitä – kehyksille täytyy määritellä myös sisältö. Kehyksen sisältö määritellään komennolla **frame**. Nämä komennot sijoitetaan **frameset**-komennon aloitus- ja lopetuskomennon väliin. **Frame**-komentoja lisätään yhtä monta kuin **frameset**-komennossa määriteltiin kehyksiä.

Kehyksen sisältö haetaan aina eri tiedostosta kuin varsinainen kehyssivu. Kun teet kehyksiä sisältävät sivun, sinun tuleekin tehdä uusi tiedosto jokaista kehystä varten. Jos teet esimerkiksi sivun, jossa on vasemmalla navigointipainikkeita sisältävä kehys, oikealla sivun sisältävä kehys ja ylhäällä otsikon sisältävä kehys, tarvitset kaiken kaikkiaan neljä tiedostoa:

- **Frame**-komennot sisältävän sivun
- navigointipainikkeet sisältävän sivun
- varsinaisen tekstin sisältävän sivun ja
- otsikon sisältävän sivun.

Oletetaan, että sivusi sijaitsevat tiedostoissa **index.html** (kehyskomennot), **navigointi.html**, **teksti.html** ja **otsikko.html**.

Aluksi tehdään kolme HTML-tiedostoa, joista tulee sivun sisältö. Ensimmäinen tiedosto on **navigointi.html**, johon tulee aikanaan navigointilinkkejä - tiedoston sisältö näyttää tältä:

```
<html>
<head><title>Navigointi</title></head>
<body>
<p>Tähän tulee navigointipainikkeita...
<ul>
<li>Eka kohde
<li>Toka kohde
<li>Kolmas kohde
</ul>
</body>
</html>
```

Seuraava tiedosto on **otsikko.html**, joka sisältää otsikon:

```
<html>
<head><title>Otsikko</title></head>
<body>
<h1>Otsikko</h1>
</body>
</html>
```

Sitten tehdään tiedosto **teksti.html**, johon sijoitetaan sivun varsinaisen asian sisältävään kehykseen tuleva sivu:

```
<html>
<head><title>Teksti</title></head>
<body>
<p>Hei, ja tervetuloa kehyksiä sisältävälle sivulleni.
<p>Näillä sivuilla testailen kehyksien rakentamista.
</body>
</html>
```

Itse tiedostoon **index.html** sijoitetaan pelkästään **frameset**- ja **frame**-komennot. Muut tiedostot (navigointi, teksti ja otsikko) ovat siis normaaleja

HTML-tiedostoja, jotka voivat sisältää mitä tahansa HTML-komentoja. Tiedostot voivat olla jo ennestään käytössä – niihin ei tarvitse tehdä muutoksia kehyksien kanssa käytettäessä.

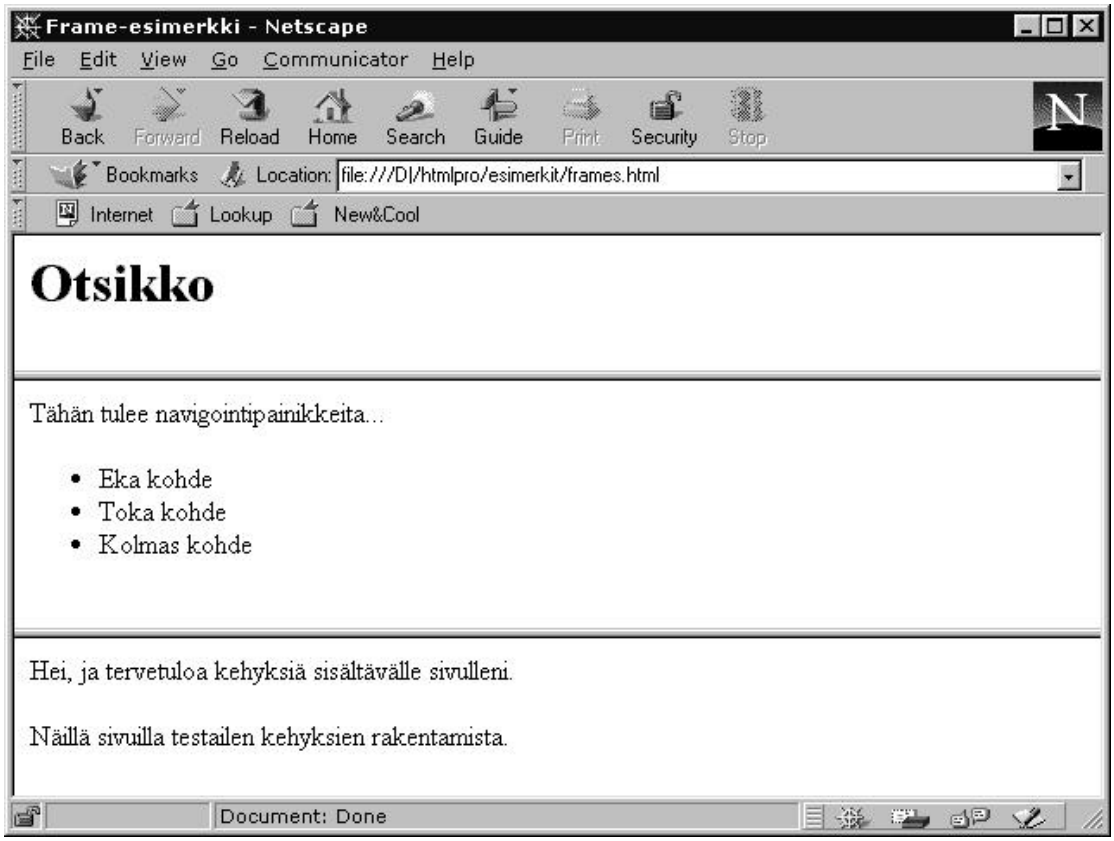
Frameset-komennossa määriteltiin päällekkäin olevat kehykset attribuutilla **rows**. Kehyksen koko pikseleinä annetaan kirjoittamalla määrittelyyn pelkkä numero. Suhteellinen koko annetaan kirjoittamalla prosenttiluku, jonka perässä on prosenttimerkki (**%**). Jos kehykselle halutaan antaa niin paljon tilaa kuin jää jäljelle, koko määritellään tähtimerkillä (*****). Kokomäärittelyt erotetaan toisistaan pilkuilla.

Frame-komento kertoo itse kehyksen sisällön. Sen attribuutissa **src** kerrotaan joko sivun URL tai sen tiedostonnimi, aivan kuten linkkiä tehtäessä.

Tehdään esimerkkisivuun kolme päällekkäin olevaa kehystä, joista ylimmäinen, otsikon sisältävä kehys on kooltaan 25% selaimen ikkunasta. Keskimmäinen kehys sisältää navigointipainikkeet ja on korkeudeltaan 150 pikseliä. Alimmalle kehykselle jätetään selaimen ikkunasta jäljelle jäävä tila ja sijoitetaan siihen tiedoston **teksti.html** sisältö.

```
<html>
<head><title>Frame-esimerkki</title></head>
<frameset rows="25%,150,*">
     <frame src="otsikko.html">
     <frame src="navigointi.html">
     <frame src="teksti.html">
</frameset>
<noframes>
<p>Valitettavasti ohjelmasi ei tue kehyksiä!
</noframes>
</html>
```

Selaimessa näet, miten kehykset sijoittuvat ruudulle:

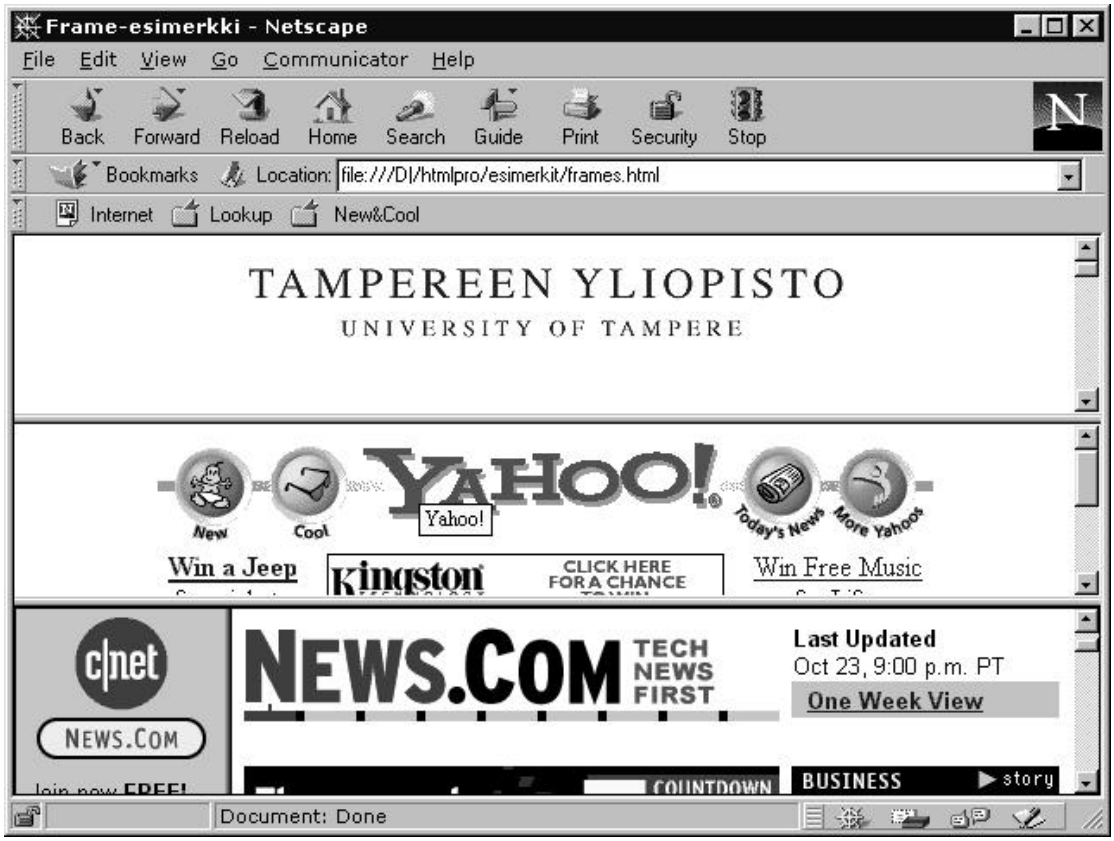


Selaimissa, jotka eivät tunne kehyksiä sivun sisältönä näkyy ainoastaan **noframes**-kohdassa annettu tiedotus.

Kehyksen sisältöä ei tarvitse välttämättä hakea samasta hakemistosta tai edes samasta koneesta kuin missä kehyssivu sijaitsee – se voi sijaita missä päin maailmaa tahansa. Voit tehdä esimerkiksi sivun, jonka eri kehykset ladataan eri palvelimista:

```
<html>
<head><title>Frame-esimerkki</title></head>
<frameset rows="33%,33%,*">
     <frame src="http://www.uta.fi/">
     <frame src="http://www.yahoo.com/">
     <frame src="http://www.news.com/">
</frameset>
<noframes>
<p>Valitettavasti ohjelmasi ei tue kehyksiä!
</noframes>
</html>
```

Näet selaimessasi kuinka eri kehysten sisällä latautuvat eri palvelimien etusivut:



Kun haluat tehdä monimutkaisempia kehysrakenteita, voit lisätä ylimääräisiä **frameset**-komentoja kehyksen määrittelyn paikalle. Tällä tavalla voit esimerkiksi jakaa ikkunan ensin pystysuunnassa kehyksiksi ja sitten paloitella vielä yksittäisiä kehyksiä vaakasuunnassa.

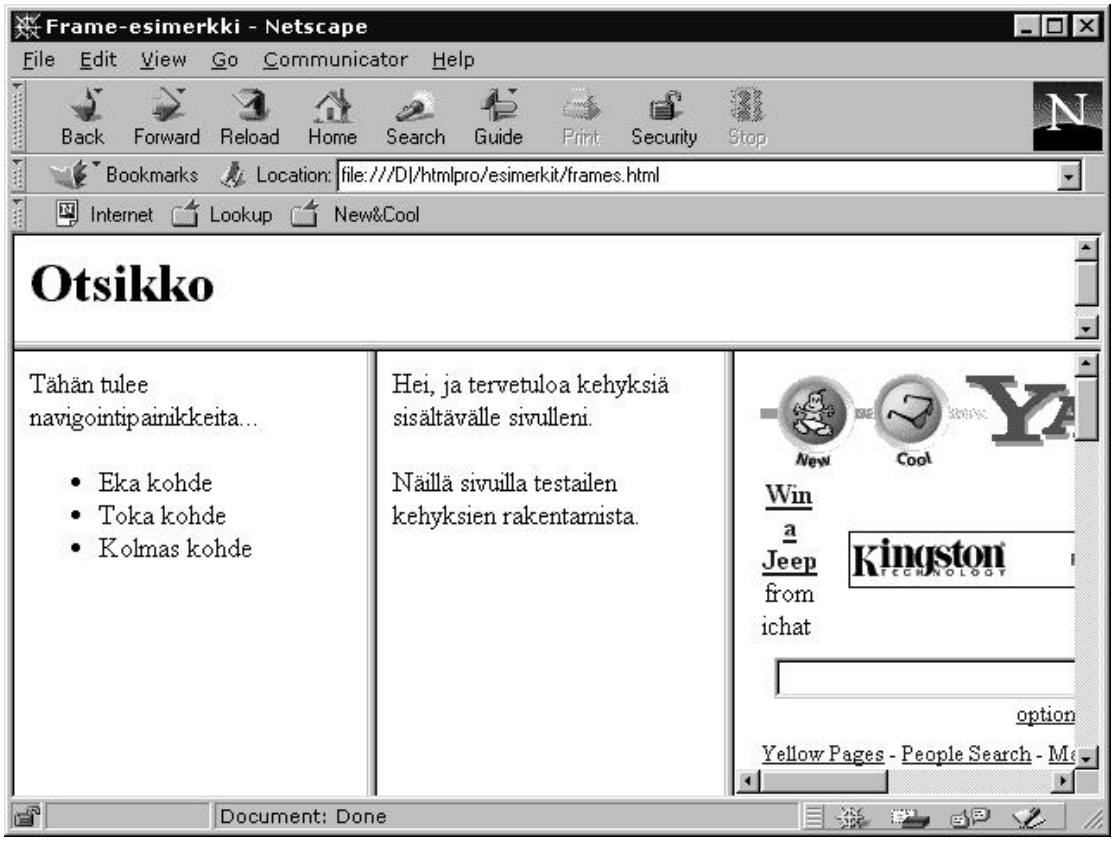
Seuraavassa esimerkissä jaetaan näyttä ensin pystysuunnassa kahteen kehykseen. Näyttö jaetaan suhteellisilla komennoilla siten, että ylempi kehys saa yhden osan ja alempi neljä osaa ruudusta. Suhteelliset osat voidaan antaa määrittelemällä kooksi osien lukumäärä ja tähtimerkki. Neljä osaa määriteltäisiin siis antamalla kooksi **4***. Kun osia on kaiken kaikkiaan viisi (4+1), alemman osan kooksi tulee 80% koko näytön korkeudesta.

Alempi kehys jaetaan vielä lisäksi kolmeen yhtä suureen osaan vaakasuunnassa. Se tapahtuu sijoittamalla komentoihin **frame**-komennon sijasta uusi **frameset**-komento, jossa määritellään mainitut kolme kehystä:

```
<frameset rows="1*,4*">
   <frame src="otsikko.html">
   <frameset cols="33%,33%,*">
      <frame src="navigointi.html">
      <frame src="teksti.html">
      <frame src="http://www.yahoo.com/">
```

```
    </frameset>
</frameset>
```

Kuvaruudulla rakenne näyttää tältä:



KEHYKSIEN OMINAISUUDET

Sivun lukija voi muokata kuvaruudulla näkyvien kehyksien kokoa tarttumalla hiirellä kehyksen reunasta ja siirtämällä sitä haluamaansa suuntaan. Jos tahdot, että kehykset pysyvät aina samankokoisina eikä lukija voi muokata sivusi sommittelua, voit lisätä kehyskomentoon attribuutin **noresize**:

```
<frame src="osoite" noresize>
```

Komento vaikuttaa myös viereisiin kehyksiin – niitäkään ei voi tämän jälkeen laajentaa sivulta, joka on yhteydessä määrittelemääsi kehykseen.

Jos kehykseen tarkoitettu sivu ei mahdu kokonaan kehyksen sisälle, kehyksen reunoihin tulevat automaattisesti näkyviin vierityspalkit. Voit kyt-

keä halutessasi vierityspalkit pois lisäämällä **frame**-komentoon attribuutin **scrolling=no**. Pidä kuitenkin huolta, että kehys on silloin tarpeeksi suuri sivuasi varten. Voit myös määritellä kehykseen näkyviin vierityspalkit silloinkin, kun ne eivät ole tarpeen. Se tapahtuu attribuutilla **scrolling=yes**. Oletusarvoinen attribuutti (vierityspalkit vain tarvittaessa) on **scrolling=auto**.

Voit määritellä kehyksen sisälle haluamasi kokoiset marginaalit lisäämällä **frame**-komentoon attribuutit **marginwidth** ja/tai **marginheight**. Marginaalien koko ilmoitetaan pikseleinä. Esimerkki:

```
<frame src="…" marginwidth=40 marginheight=30>
```

Kun suunnittelet sivua, jossa käytetään kehyksiä, kokeile sen toimintaa erikokoisilla ikkunoilla. Kaikki käyttäjät eivät käytä selaintaan samankokoisessa ikkunassa – joillakuilla selaimen ikkuna voi olla tilantarpeen vuoksi kutistettuna vain neljännesruudun kokoiseksi, toisilla taas venytettynä koko ruudun kokoiseksi. On myös käyttäjiä, joiden kuvaruudun resoluutio eroaa paljon juuri sinun käyttämästäsi resoluutiosta.

LINKKIEN TOIMINTA KEHYKSISSÄ

Kehyksiä sisältävää sivua tehtäessä on tärkeää suunnitella sivun käyttöliittymä toimivaksi ja selkeäksi. Kehyssivulla on usein tarpeen tehdä kehyksiä, jotka vaikuttavat toistensa sisältöön.

Esimerkkinä voisi olla vaikka sivu, jonka vasemmassa reunassa on navigointipalkki ja oikeassa reunassa varsinainen sivu. Kun navigointipalkista näpäytetään jotain linkkiä, oikealla näkyvään alueeseen ladataan haluttu sivu.

Tällainen sivu toteutetaan antamalla kehyksille tunnisteet, joita voidaan käyttää linkin kohteina. Kehykselle annetaan tunniste lisäämällä **frame**-komentoon attribuutti **name**.

Tehdään esimerkiksi ylläkuvatun kaltainen sivu. Vasempaan reunaan tehdään navigointipalkki, jonka koko on 20 prosenttia ruudun leveydestä. Loppuosa ruudusta jätetään varsinaiselle sivulle määritellyksi alueeksi. Tälle alueelle määritellään tunniste, jota voidaan käyttää myöhemmin linkeissä.

```
<frameset cols="20%,*">
<frame src="linkit.html">
<frame src="sivu.html" name="dokum">
</frameset>
```

Tiedosto **sivu.html** sisältää vaikkapa tervetulotoivotuksen, joka ladataan lukijan näytölle, kun hän saapuu sivulle ensimmäistä kertaa. Tiedosto **linkit.html** sisältää linkkejä erilaisiin sivuihin, jotka on tarkoitus näyttää oikean reunan alueella.

Jos tiedosto **linkit.html** sisältää tavallisen linkin ilman lisäattribuutteja, sen näpäyttäminen lataa linkin osoittaman tiedoston vasemman reunan kehykseen, navigointipainikkeiden tilalle. Tarkoituksena on kuitenkin, että linkin valitseminen muuttaa oikean reunan aluetta. Tällaisessa tapauksessa käytetään linkkikomentoon lisättävää **target**-attribuuttia, jolla kerrotaan minne sivu ladataan. Esimerkki:

```
<a href="juttu-1.html" target="dokum">Ensimmäinen
tarina</a>
```

Yllä oleva esimerkki lataa tiedoston **juttu-1.html** kehykseen nimeltään **dokum**. Tällä nimellä määritelty kehys löytyykin kuvaruudun oikeasta reunasta. Jos kehykselle ei olisi annettu tunnistetta, tiedostoa **juttu-1.html** varten olisi avattu kokonaan uusi ikkuna.

Jos haluat linkin latautuvan koko ruudun kokoiseksi (ja poistavan vanhat kehykset), voit määritellä sen kohteeksi **target="_top"** (huomaa alleviivausmerkki).

TYYLIPOHJAT

Tyylipohjat (**Style Sheets**) ovat uusi tapa määritellä sivun ulkonäköä. Vanhempi tapa on käyttää erilaisia HTML-komentoja, jotka määrittelevät kuvaruudulla näkyvän sivun ulkoasun.

Tyylipohjien avulla on mahdollista rakentaa sivu ensin HTML-komentoja käyttäen ja sen jälkeen määritellä eri tyyleille niiden kuvaruudulla näkyvä ulkoasu. Tällöin on mahdollista esimerkiksi kertoa, että kaikki **<h1>** -komennoilla merkityt otsikot tulee näyttää kuvaruudulla punaisella värillä 18 pisteen kokoisina, tai että tekstikappaleet näytetään **Helvetica**-fontilla ja keskitetään ruudulle.

Samaa tyylipohjaa voi käyttää useissa sivuissa, jolloin on mahdollista suunnitella esimerkiksi yksi yhteinen tyylipohja kaikille tekemillesi sivuille ja kytkeä tämä tyylipohja niihin. Kun haluat muuttaa sivuja, muutos tarvitsee tehdä vain tyylipohjaan, jolloin kaikki sitä hyväkseen käyttävät sivut vaihtavat automaattisesti ulkoasuaan.

Tyylipohjissa määriteltyihin tyyleihin voi tehdä myös päällekkäisiä tyylimäärittelyitä (**Cascading Style Sheets**): sivulla voidaan antaa uusi tyylipohja, jossa annetut määrittelyt ajavat kaikille yhteisen tyylipohjan yli. Sen lisäksi tyylejä on mahdollista antaa vielä sivun sisällä, varsinaisissa HTML-komennoissa, jolloin nämä tyylit ajavat taas edellisten tyylipohjien edelle. Tällä tavalla on mahdollista toteuttaa erilaisia poikkeuksia.

Tulevissa selaimissa käyttäjä voi tehdä myös oman tyylipohjansa, jonka avulla hän määrittelee tietyt säännöt, jotka ajavat verkon sivuilta löytyvien sääntöjen yli. Värisokeat käyttäjät voivat määritellä eri tyyleille itselleen parhaiten sopivat värit ja heikkonäköiset voivat määrätä tekstikoot normaalia isommiksi.

TYYLIPOHJIEN KOMENNOT

Tyylimäärittelyt (**Style Definition**) kertovat, miltä tietyntyyppisen tekstin tulee näyttää näytöllä. Jos haluat esimerkiksi määritellä ykköstason otsikot väriltään punaisiksi ja kooltaan 30 pisteen fontilla esitettäviksi, voit käyttää seuraavanlaista tyylimäärittelyä:

```
H1    { font-size: 30pt;
        color: red }
```

Määrittelyissä annetaan ensin HTML-komento, johon määrittelyt vaikuttavat. Sen jälkeen annetaan kaarisulkujen sisällä joukko puolipisteellä erotettuja määrittelyitä, jotka kertovat kyseisen tyylin ruudulla näkyvän ulkoasun.

Jos haluat määritellä koko sivun ulkoasun kerralla, voit kytkeä tyylimäärittelyt **body**-komentoon. Sen jälkeen voit täsmentää tarkemmin niitä komentoja, joiden haluat tulostuvan eri tavalla:

```
/* Kaikki teksti tulostetaan mustalla */
/* 18 pisteen normaalifontilla... */
BODY  {  font-size: 18pt;
         color: black;
         font-weight: medium }
/* …paitsi ykkötason otsikot, joiden */
/* koko on 20 pistettä. */
H1    {  font-size: 20pt }
```

Kuten huomaat, tyylimäärittelyiden sisään voi sijoittaa kommentteja merkkien **/*** ja ***/** väliin. Edellisessä esimerkissä sivun tekstityyliksi määriteltiin musta 18 pisteen normaalikokoinen kirjasintyyppi. Kaikki sivulla olevat tekstityylit (otsikot, leipäteksti jne.) sijaitsevat **body**-komennon sisällä, joten ne perivät **body**-komennolle annetut määrittelyt. Tyylimäärittelyissä on mahdollista antaa tarkempia määrittelyitä, jolloin tarkennukset ajavat perittyjen määrittelyiden yli. Esimerkissä ykköstason otsikoiden (**h1**) ulkoasulle onkin annettu tarkennuksena uusi fonttikoko, 20 pistettä.

Jos haluat antaa useammalle tyylille määrittelyt yhtäaikaa, voit antaa näiden määrittelyt yhdessä:

```
P, ADDRESS, H3  {  font-family: "Arial";
                   font-style: italic;
                   font-weight: bold }
```

Tämä määrittely tekee leipätekstistä, osoitekappaleista ja kolmostason otsikoista lihavoitua ja kursiivia **Arial**-tekstiä. Tekstikoko tulee joko yleisimmistä tyylimäärittelyistä, kerrotaan myöhemmin tai selain käyttää oletuskokoja.

Tyylimäärittelyissä voidaan käyttää useita erilaisia yksiköitä. Esimerkiksi fonttikoko on mahdollista määritellä pistekokona, tuumina, senttimetreinä, pikseleinä tai prosenttiarvona:

```
{ font-size: 12pt }
/* 12 pisteen fontti */

{ font-size: 3in}
/* 3 tuumaa korkea fontti */

{ font-size: 3cm}
/* 3 senttiä korkea fontti */

{ font-size: 50px }
/* 50 pikseliä korkea fontti */

{ font-size: 200% }
/* 2 kertaa perusfonttia suurempi */
```

Tyylimäärittelyyn on mahdollista tuoda aiempia tyylimäärittelyitä. Tällä tavalla määrittelyt voidaan koota useasta lähteestä, jossa kopioituja määrittelyitä yhdistellään ja määritellään vielä entistä tarkemmin. Aiemmin määritelty tyylitiedosto voidaan tuoda mukaan komennolla **@import**:

```
@import url(yleinen.css);
H1   { color: red }
/* Tarkennetaan määrittelyä */
```

Selain lukee ensin määrittelyt tyylitiedostosta **yleinen.css** (nimen paikalle voidaan antaa myös URL) ja täydentää sieltä ladattuja tyylejä muuttamalla ykköstason otsikot punaisiksi.

TYYLITIEDOSTOT

Tyylimäärittelyt voidaan kirjoittaa omaan tiedostoonsa ja kytkeä yhteen tai useampaan HTML-tiedostoon. Kun teet tyylitiedostoa, anna sille tiedostonnimen tarkentimeksi **.css**. Se tulee englanninkielen sanoista **Cascading Style Sheet**: päällekkäiset tyylimäärittelyt.

Tyylitiedosto kytketään WWW-sivun **<head>**-osan sisään komennolla **<link>**. Komennolle annetaan lisämäärittelyinä tyylitiedoston sijainti attribuutilla **href** sekä sen tyyppi attribuutilla **type**. Tyypiksi annetaan aina **"text/css"**, joka erottaa **css**-tyyppisen tyylimäärittelytiedoston mahdollisista muilla tavoilla toteutetuista tyylimäärittelyistä. Lisäksi komentoon annetaan attribuutti **rel=stylesheet**, joka kertoo lisättävän asian olevan juuri tyylitiedosto.

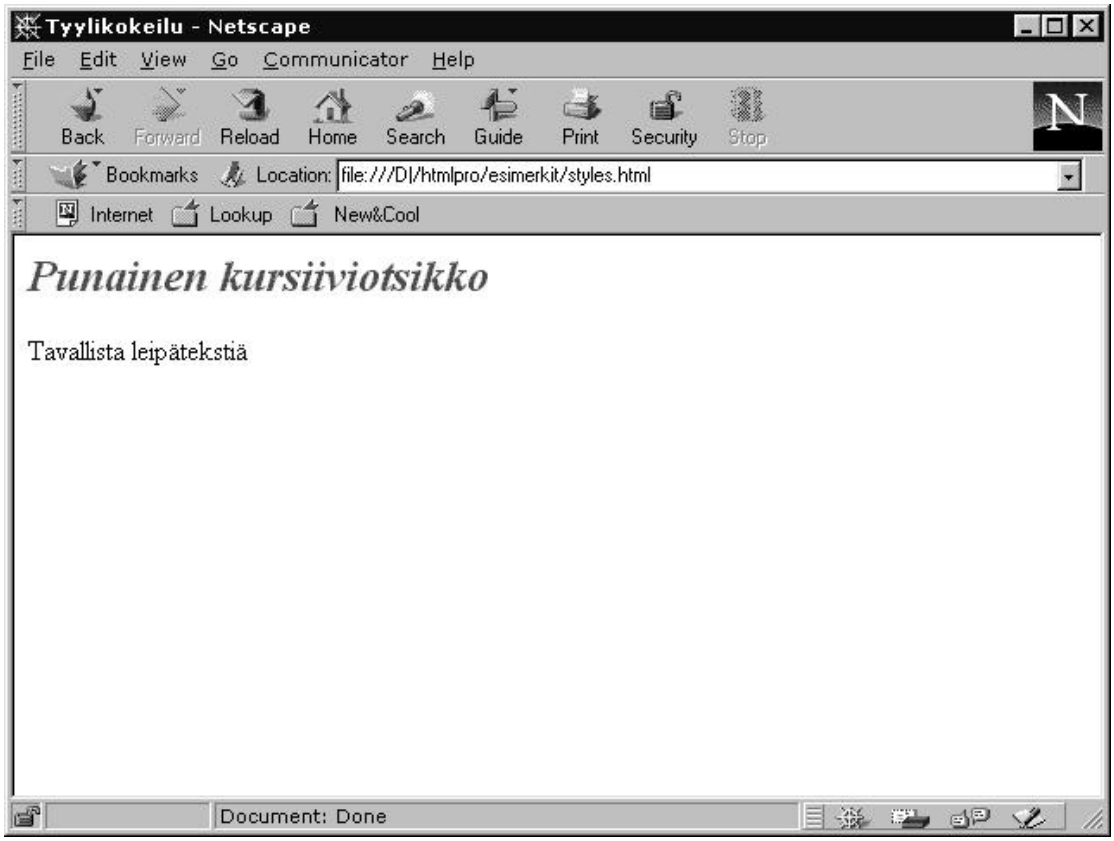
Tehdään esimerkiksi seuraavanlainen tyylitiedosto:

```
H1  { font-size: 20pt;
      font-style: italic;
      color: red }
```

Tallennetaan se tiedostoon **tyyli.css**. Seuraavaksi tehdään sivu, johon tämä tyylitiedosto kytketään:

```
<html>
<head>
<title>Tyylikokeilu</title>
<link rel=stylesheet type="text/css" href="tyyli.css">
</head>
<body>
<h1>Punainen kursiiviotsikko</h1>
<p>Tavallista leipätekstiä
</body>
</html>
```

Näet selaimessa tyylitiedoston vaikutuksen:



Kun teet muutoksia tyylitiedostoon, ne vaikuttavat automaattisesti kaikkiin tiedostoa hyväksi käyttäviin HTML-tiedostoihin. Siten on mahdollista tehdä yksi yhtenäinen tyyli kaikille palvelimessa sijaitseville sivuille ja muuttaa niiden kaikkien ulkoasua vain yhtä tiedostoa muokkaamalla.

TYYLIMÄÄRITTELYT HTML-TIEDOSTOSSA

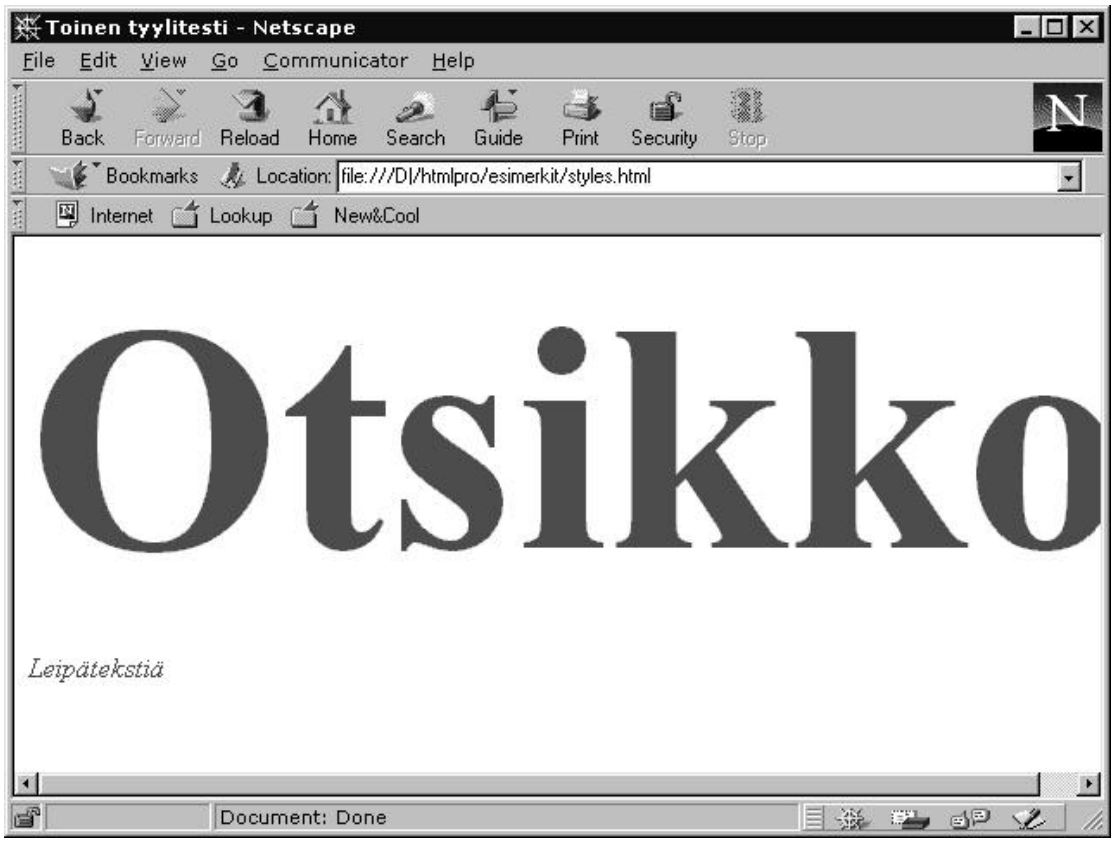
Tyylimäärittelyt on myös mahdollista sijoittaa HTML-tiedoston sisään. Sinne voi sijoittaa kokonaisia tyylitiedostoja, mutta tavallisempi tapa on antaa tarkennuksia joillekin tietyille tyyleille vain kyseistä sivua varten.

Tyylimäärittelyt sijoitetaan HTML-tiedostoon, **<head>** -osan sisään. Määrittelyt ympäröidään **<style>** ja **</style>** -komennoilla. Aloituskomentoon annetaan attribuutti **type**, joka kertoo annettavien tyylimäärittelyiden tyypin: normaalisti tähän annetaan **type="text/css"** joka kertoo tyylimäärittelyiden olevan **css**:n mukaisia.

Jotta tyylimäärittelyt eivät tulisi vanhemmissa selaimissa näkyviin leipätekstin seassa, niiden ympärille kannattaa sijoittaa HTML:n kommenttimerkit. Alla on esimerkki tyylimäärittelyiden antamisesta sivun sisällä:

```
<html>
<head>
<title>Toinen tyylitesti</title>
<style type="text/css">
<!--
  BODY   { color: green }
  H1     { font-size: 5cm;
           color: red }
  P      { font-style: italic }
-->
</style>
</head>
<body>
<h1>Otsikko</h1>
<p>Leipätekstiä</p>
</body>
</html>
```

Esimerkissä kuvaruudulle tulee viiden senttimetrin korkuinen punainen otsikko ja vihreää kursivoitua leipätekstiä:



TYYLIMÄÄRITTELYT KOMENNOISSA

Myös itse HTML-komentoihin on mahdollista antaa tyylimäärittelyitä. Ne ajavat jälleen sivuun kytkettyjen ja alussa määriteltyjen tyylien yli. Voit antaa HTML-komennolle tyylimäärittelyitä lisäämällä komentoon attribuutin **style**:

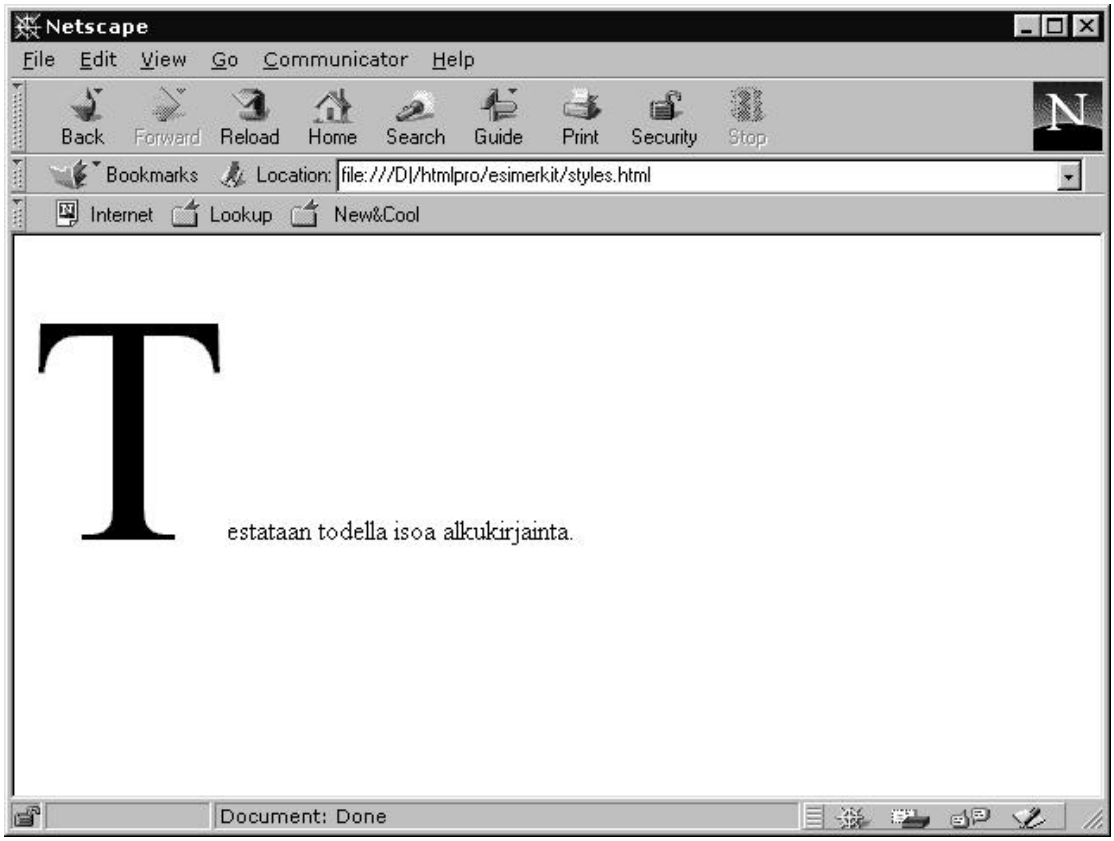
```
<p>Normaalia leipätekstiä</p>
<p style="color: red">Punaista leipätekstiä</p>
<p>Taas normaalia leipätekstiä.</p>
```

Jos haluat antaa tyylimäärittelyitä useamman kuin yhden komennon mittaiselle alalle, voit antaa tyylimäärittelyt ** ** -komennon avulla. Ympäröi tällöin haluamasi alue ** **-komennoilla ja sijoita tyylimäärittelyt tämän komennon yhteyteen:

```
<div style="margin-left: 5cm; margin-right: 5cm">
<h1>Marginaalit</h1>
<p>Div-komentojen väliin sijoitettu osa dokumentista
tulee näkyviin 5 senttimetrin marginaaleilla
vasemmassa ja oikeassa reunassa.</p>
</div>
<p>Ja tämän jälkeen taas normaaleja marginaaleja.
```

Kolmas tapa on käyttää komentoa ****. Tämä komento toimii samalla tavalla kuin ** ** -komentokin, mutta on tarkoitettu lyhyemmille, sanojen tai jopa merkkien mittaisille väleille. Esimerkki:

```
<p><span style="font-size: 5cm">T</span>estataan
todella isoa alkukirjainta.</p>
```



TYYLILUOKAT

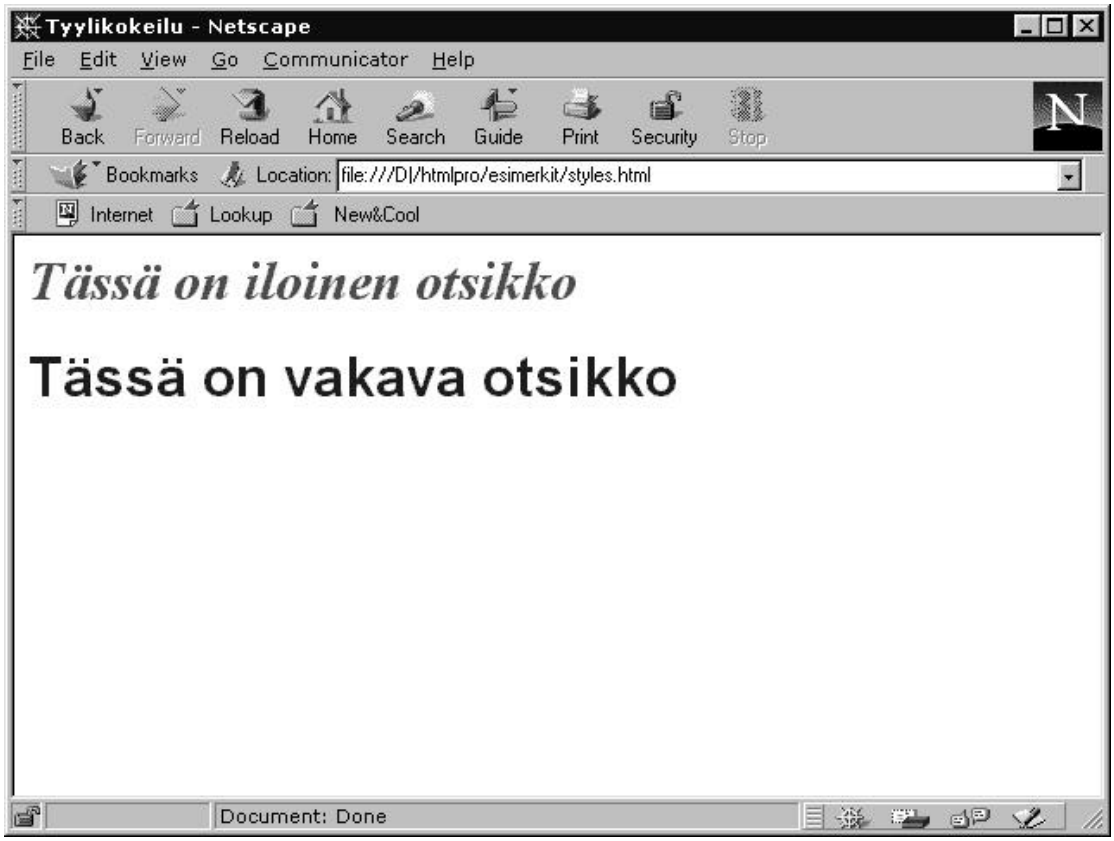
Voit tehdä yhdelle HTML-komennolle useampia ennalta määriteltyjä tyylejä ja erotella ne jo valmiiksi tyylitiedostossa. Näitä ennalta määriteltyjä tyylejä kutsutaan luokiksi.

Tehdään esimerkiksi ykköstason otsikoille kaksi ennalta määriteltyä tyyliä: "iloinen", jossa teksti on kirkkaanpunaista ja kursivoitua; ja "vakava", jossa teksti on **Arial**-tekstityylillä tulostettua ja valkoista. Määrittelyt tapahtuvat silloin seuraavasti:

```
H1.iloinen{ color: red;
            font-style: italic }
H1.vakava { font-family: "Arial";
            color: blue }
```

Kun olet kytkenyt määrittelyt HTML-tiedostoon, voit käyttää sen sisällä ennalta määriteltyjä tyylejä lisäämällä komentoon attribuutin **`class`**:

```
<h1 class="iloinen">Tässä on iloinen otsikko</h1>
<h1 class="vakava">Tässä on vakava otsikko</h1>
```



Joillekin HTML-komennoille on olemassa valmiiksi määriteltyjä luokkia, nk. pseudoluokkia (**pseudo-classes**). Esimerkiksi linkkien ulkoasua määriteltäessä on pystyttävä erottelemaan sellaiset linkit, joissa käyttäjä on käynyt ennen ja sellaiset linkit, joissa ei ole vielä käyty. Tämä tapahtuu pseudoluokkien avulla. Pseudoluokkien määrittely tapahtuu samalla tavalla kuin tavallistenkin luokkien - erona on vain kaksoispiste tavallisen pisteen sijasta:

```
A:link    { color: yellow }
A:visited { color: green }
```

Edellisessä esimerkissä määritellään jo nähdyt linkit vihreiksi ja käymättömät keltaisiksi.

TYYLIMÄÄRITTELYJEN KOMENNOT

Alla esitellään nykyisten selaimien ymmärtämät tyylimäärittelyt. Tyylimäärittelyitä on tässä mainittujen lisäksi myös lisää - niistä löydät lisätietoja **W3C**:n sivuilta, osoitteesta

http://www.w3.org/

Tässä lähteenä käytetty määrittely löytyy osoitteesta:

http://www.w3.org/TR/REC-CSS1

TEKSTITYYPIT

Tekstityypin ulkoasua voit säätää seuraavilla määrittelyillä:

font-size

Tällä määritellään tekstin koko. Voit antaa koon pisteinä (**pt**), tuumina (**in**), senttimetreinä (**cm**), pikseleinä (**px**) tai prosenttikokona (**%**). Esimerkki: **font-size: 12pt**

font-family

Määrittelee fontin. Voit antaa yhden tai useampia fontin nimiä, jolloin selain voi poimia listalta uuden fontin, jos se ei löydä listan ensimmäistä. Fontin nimeksi voi antaa myös jonkin fonttityylin yleisnimistä (**serif**, **sans-serif**, **cursive**, **fantasy** tai **monospace**). Jos fontin nimessä on välilyöntejä, kirjoita se lainausmerkkien sisään. Esimerkki: **font-family: "Times New Roman", Times, serif**

font-weight

Määrittelee fontin paksuuden. Tarjolla on useita vaihtoehtoja: **extra-light**, **light**, **demi-light**, **medium**, **demi-bold**, **bold** ja **extra-bold**. Nykyiset selaimet tukevat normaalisti vain **mediumia** (normaali) ja **boldia**. CSS tarjoaa lisäksi arvot **normal**, **bold**, **bolder**, **lighter** ja numerot 100-900 sadan välein. Esimerkki: **font-weight: bold**

font-style

Määrittelee fontin tyylin. CSS tarjoaa vaihtoehdot **normal**, **italic** ja **oblique**. Nykyiset ohjelmat tukevat arvoja **normal** ja **italic**. Esimerkki: **font-style: italic**

line-height

Määrittelee rivinvälin. Voit määritellä rivinvälin kuten fontin korkeudenkin: pisteinä, pikseleinä, senttimetreinä, tuumina tai prosenttiarvona. Esimerkki: **line-height: 150%**

text-decoration

Määrittelee tekstin ulkoasun. Vaihtoehtoja ovat **underline** (alleviivaus), **overline** (päälleviivaus), **line-through** (yliviivaus) ja **blink** (vilkkuminen). Internet Explorer 3.0 ei tue vilkkumista eikä päälleviivausta. Esimerkki: **text-decoration: underline**

VÄRIT

Värejä ja kuvioita voit säätää seuraavilla määrittelyillä:

color

Määrittelee värin. Voit käyttää samoja värien nimiä tai väriarvoja kuin **<font>** -komentoa käyttäessäsi tai **RGB**-arvoja. Esimerkki: **color: red** ja **color: rgb(255,0,0)**. Molemmilla määrittelyillä saat punaista väriä.

background

Määrittelee taustavärin tai -kuvan. Sen avulla voit antaa taustavärin koko sivun lisäksi myös yksittäisille elementeille (kun asetat tekstille taustavärin, teksti näyttää ruudulla kuin värikynällä alleviivatulta). Voit antaa taustavärin joko värinä tai väriarvona, kuten yllä, tai jos haluat tehdä taustakuvan, voit antaa sen osoitteen muodossa **URL(*osoite*)**. Sulkujen sisään tulee siis haluamasi kuvan osoite tai tiedostonnimi.

Jos määrittelet taustakuvan, voit määritellä lisäksi, toistetaanko taustakuvaa. Määrittely **repeat** saa taustakuvan käyttäytymään aivan kuten HTML-sivun taustakuvan: sitä toistetaan vierekkäin ja allekkain, kunnes se täyttää toivotun alueen. Määrittely **repeat-x** toistaa kuvaa vain vaakasuunnassa ja **repeat-y** vain pystysuunnassa. Määrittely **no-repeat** kertoo, että kuvaa ei tule toistaa.

Määrittely **scroll** kertoo, että taustakuvan tulee vieriä tekstin mukana. Jos haluat, että taustakuva pysyy paikallaan samalla kuin teksti vierii, käytä määrittelyä **fixed**.

Kuvan paikan pystyy säätämään antamalla sen sijainnin kuvaruudulla prosentteina, jonain muuna mittayksikkönä (kuten fonttikoossa) tai sanallisena sijaintina (**top**, **middle**, **bottom**, **left**, **center**, **right**).

Esimerkkejä:

```
background: red
background: URL(http://osoite/t.gif) scroll no-repeat
background: URL(http://osoite/t.gif) top right
/* kuva sijoitetaan oikeaan yläkulmaan */
background: URL(http://osoite/t.gif) 0% 100%
/* sama numeroina */
```

MARGINAALIT JA TEKSTIN SIJOITTELU

Voit säätää marginaaleja määrittelyillä **margin-left** (vasen), **margin-right** (oikea) ja **margin-top** (ylämarginaali). Marginaalin koon voi antaa pikseleinä, pisteinä, senttimetreinä ja tuumina (kuten fonttimäärittelyissä). Marginaaleille voi antaa myös negatiivisia arvoja, jolloin marginaali siirtyy toiseen suuntaan. Esimerkki: **margin-left: -3cm**

Tekstiä voi sisentää määrittelyllä **text-indent**. Mittayksiköt ovat samoja kuin marginaaleissakin. Negatiivinen sisennys vie tekstiä marginaalin sisään. Esimerkki: **text-indent: 1cm**

Tekstin asemointiin voi vaikuttaa määrittelyllä **text-align**. Mahdolliset arvot ovat **left** (vasempaan laitaan tasattu), **right** (oikealle tasattu) ja **center** (keskitetty). Esimerkki: **text-align: right**

TASOT

Jos haluat määritellä mahdollisimman tarkasti selaimen ruudulla näkyvän sivun ulkoasun, voit käyttää sijoitteluun tasomäärittelyitä.

Tasomäärittelyitä voidaan tehdä joko HTML-komennoilla tai tyylimäärittelyiden avulla. Niiden avulla voit määritellä minkä tahansa sivun osan paikan pikselin tarkkuudella.

Voit kuvitella tason (**layer**) ikkunaksi, joka sisältää HTML-koodia. Voit kertoa tasolle sen paikan selaimen näytöllä ja sen koon. Taso tulee näkyviin sivun normaalin sisällön päälle tai alle. Voit sijoittaa ikkunaan useita tasoja. Jos tasot menevät päällekkäin, voit määritellä mitkä tasot näkyvät toisten tasojen päällä.

Huomaa, että tasot toimivat vain uusimmissa selainversioissa (Netscape 4.0 ja uudemmat sekä Internet Explorer 4.0 ja uudemmat). Jos aiot tehdä sivuillesi tasoja, tee sivuistasi sellaiset, että niitä voi lukea myös vanhemmilla selainversioilla. Internet Explorerissa toimivat tätä kirjoitettaessa vain tyylimäärittelyillä toteutetut tasot.

TASOT HTML-KOMENNOILLA

Voit lisätä selaimen ikkunaan tason komennolla **`layer`**. Komennolle annetaan attribuutteja, jotka kertovat tason paikan, sen koon ja siinä käytetyt värit.

Tason koko määritellään attribuuteilla **`width`** ja **`height`**. Attribuutit voivat saada arvokseen pikselimäärän tai prosenttikoon selaimen ikkunasta. Tason sijainti selaimen näytöllä määritellään attribuuteilla **`top`** ja **`left`**, jotka kertovat tason vasemman yläkulman paikan.

Alla näet esimerkin tavallisesta WWW-sivusta, jonka päälle on lisätty keltainen taso **`layer`**-komennolla.

```
<h1>Normaali sivu</h1>
<p>Tämä teksti on normaalia WWW-sivulla olevaa
tekstiä.
```

```
<p>Sen ylle on lisätty taso layer-komennolla.

<layer top="25%" left="25%" width=200 height=100
bgcolor="Yellow">
<h1>Taso</h1>
<p>Tämä teksti sijaitsee tason sisällä!
</layer>

<p>Tämä teksti kuuluu jälleen tavalliseen sivuun.
```

Tason vasemman yläkulman paikaksi on määritelty neljännes (25%) sivun yläreunasta ja neljännes sivun vasemmasta reunasta. Tason koko on annettu pikseleinä: sen leveys on 200 pikseliä ja korkeus 100 pikseliä. Tällä tavalla taso tulee tavallisella sivulla olevan tekstin päälle ja peittää sen osittain. Tason taustaväriksi määriteltiin keltainen, joten alla oleva teksti ei näy tason pohjan läpi:

Tason sisään voidaan sijoittaa normaalia HTML-koodia, kuten tekstiä, kuvia ja taulukoita.

Attribuutit **top** ja **left** eivät ole pakollisia - jos jätät ne pois, taso ilmestyy siihen paikkaan, jossa taso määriteltiin. Voit jättää tasosta pois myös leve-

ys- ja korkeusmäärittelyt, jolloin selain määrittelee itse tason koon. Taso jatkuu silloin vasemmalle ja alas aina selaimen ikkunan reunoihin saakka.

Tason taustaväri määritellään attribuutilla **bgcolor**, aivan kuten **body**-komennossa. Tasolle voidaan määritellä myös taustakuva attribuutilla **background**.

Tasoja voidaan asetella myös sisäkkäin. Kun olemassa olevan tason sisään sijoitetaan toinen taso, sen sijainti määritellään edellisen tason vasemmasta yläkulmasta lähtien. Esimerkki:

```
<h1>Normaali sivu</h1>
<p>Tämä teksti on normaalia WWW-sivulla olevaa
tekstiä.
<p>Sen ylle on lisätty taso layer-komennolla.
<layer top="25%" left="25%" width=200 height=100
background="grey_dots.gif" id="taso1">
   <h1>Taso</h1>
   <p>Tämä teksti sijaitsee tason sisällä!
   <layer id="taso2" top="20" left="100"
bgcolor="Yellow">
        <p>Vielä toinenkin taso!
   </layer>
</layer>
<p>Tämä teksti kuuluu jälleen tavalliseen sivuun.
```

Esimerkki sisältää kaksi tasoa, jotka on sijoitettu sisäkkäin. Ensimmäisenä määritellään taso, jonka vasen yläkulma on neljänneksen selaimen ikkunan leveydestä etäisyydellä WWW-sivun yläreunasta ja vasemmasta reunasta. Tämän tason leveys on 200 pikseliä ja korkeus 100 pikseliä. Tason taustakuvioksi määritellään tiedostossa **grey_dots.gif** oleva kuva. Tasolle annetaan tunnus attribuutilla **id** - tällä tavalla siihen voidaan viitata myöhemmin. Tason tunnus, tai nimi, on "**taso1**".

Taso1 sisältää otsikon ja tekstikappaleen. Näiden lisäksi sen sisään sijoitetaan uusi taso, jonka tunnukseksi annetaan "**taso2**" **id**-attribuutilla. Tämän tason sijainti määrittyy tason "**taso1**" vasemman ylänurkan perusteella - taso2 sijaitsee 20 pikseliä alaspäin ja 100 pikseliä oikealle **taso1**:n nurkasta. Jotta **taso2** erottuisi paremmin, sen taustaväriksi määritellään keltainen. Tuloksen näet selaimessa:

Kun taso sijoitetaan toisen tason sisään, sen sijoituskohta ja koko annetaan suhteessa edellisen tason kokoon. Esimerkiksi attribuutti **width="25%" taso2**:ssa määrittelisi tason leveydeksi neljänneksen **taso1**:n koosta. Jos tason leveyttä ja korkeutta ei määritellä, sen koko kasvaa, kunnes se törmää edellisen tason reunoihin.

Tason sijainti voidaan määritellä myös suhteessa WWW-sivun nurkkaan. Tähän käytetään attribuutteja **pagex** ja **pagey**:

```
<h1>Normaali sivu</h1>
<p>Tämä teksti on normaalia WWW-sivulla olevaa
tekstiä.
<p>Sen ylle on lisätty taso layer-komennolla.
<layer top="25%" left="25%" width=200 height=100
background="grey_dots.gif" id="taso1">
   <h1>Taso</h1>
   <p>Tämä teksti sijaitsee tason sisällä!
   <layer id="taso2" pagex=400 pagey=200
bgcolor="Yellow">
        <p>Vielä toinenkin taso!
   </layer>
```

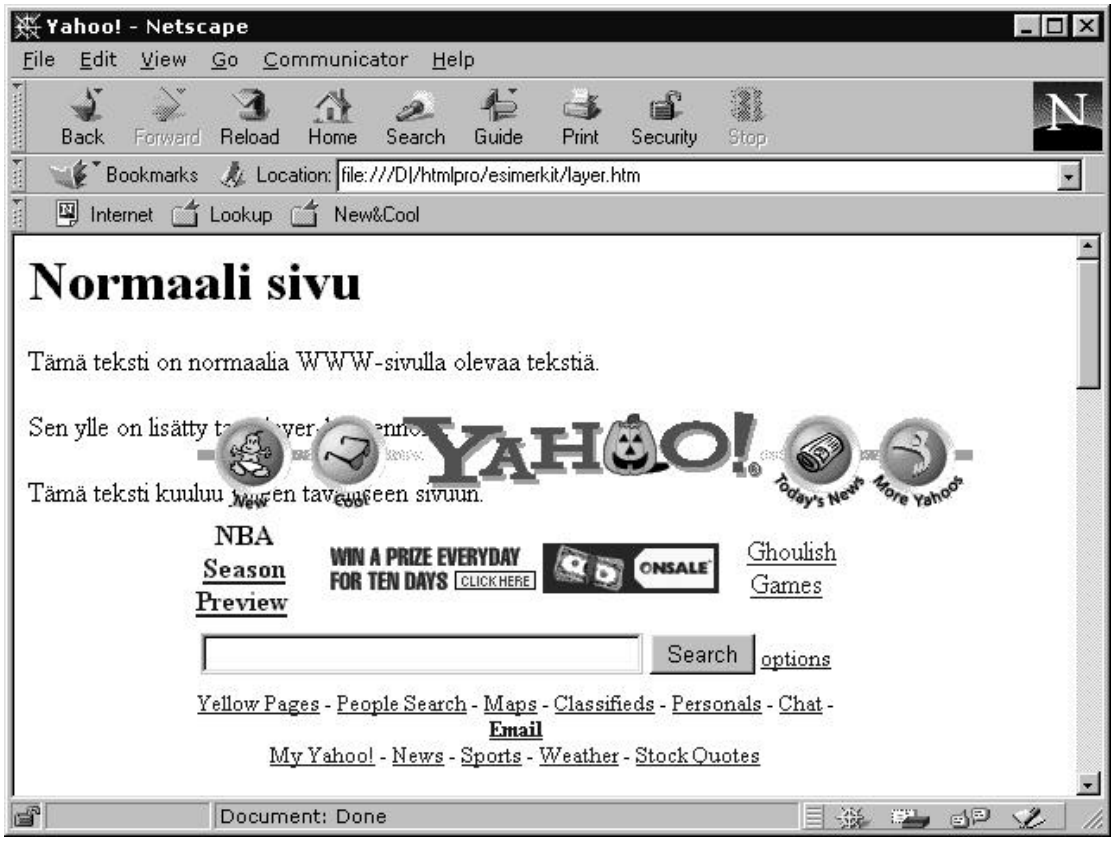
```
</layer>
<p>Tämä teksti kuuluu jälleen tavalliseen sivuun.
```

Nyt **taso2** sijoittuu kohtaan, joka on 200 pikseliä sivun yläreunasta ja 400 pikseliä vasemmasta reunasta. Koska **taso2** sijaitsee **taso1**:n sisällä, **taso1** venyy sen verran, että **taso2** tulee näkyviin ruudulle.

Tason sisältö voidaan hakea myös toisesta tiedostosta tai toisesta URL-osoitteesta. Osoite tai tiedoston nimi määritellään **src**-attribuutilla:

```
<h1>Normaali sivu</h1>
<p>Tämä teksti on normaalia WWW-sivulla olevaa
tekstiä.
<p>Sen ylle on lisätty taso layer-komennolla.

<layer top=100 left=100 width=300 height=200
src="http://www.yahoo.com" id=taso1>
</layer>

<p>Tämä teksti kuuluu jälleen tavalliseen sivuun.
```

Sivua katsottaessa siinä olevaan tasoon latautuu **Yahoon** kotisivu:



Huomaa, että koska **Yahoon** kotisivu ei mahtunut kokonaan tason sisään, se venyy pystysuunnassa niin paljon, että koko sivu mahtuu näkyviin. Tason alle jäävä teksti tulee näkyviin tasossa olevan tekstin läpi, koska **layer**-komennossa ei annettu **bgcolor**-attribuuttia. Voit tehdä tasosta läpinäkymättömän lisäämällä komentoon esimerkiksi attribuutin **bgcolor="White"**.

Voit määritellä tasot myös siten, että niistä näkyy vain osa. Tämä tapahtuu lisäämällä **layer**-komentoon attribuutti **clip**. Attribuutille annetaan arvot, jotka kertovat mikä osa tasosta tulee näyttää. Arvoja voi olla joko neljä tai kaksi.

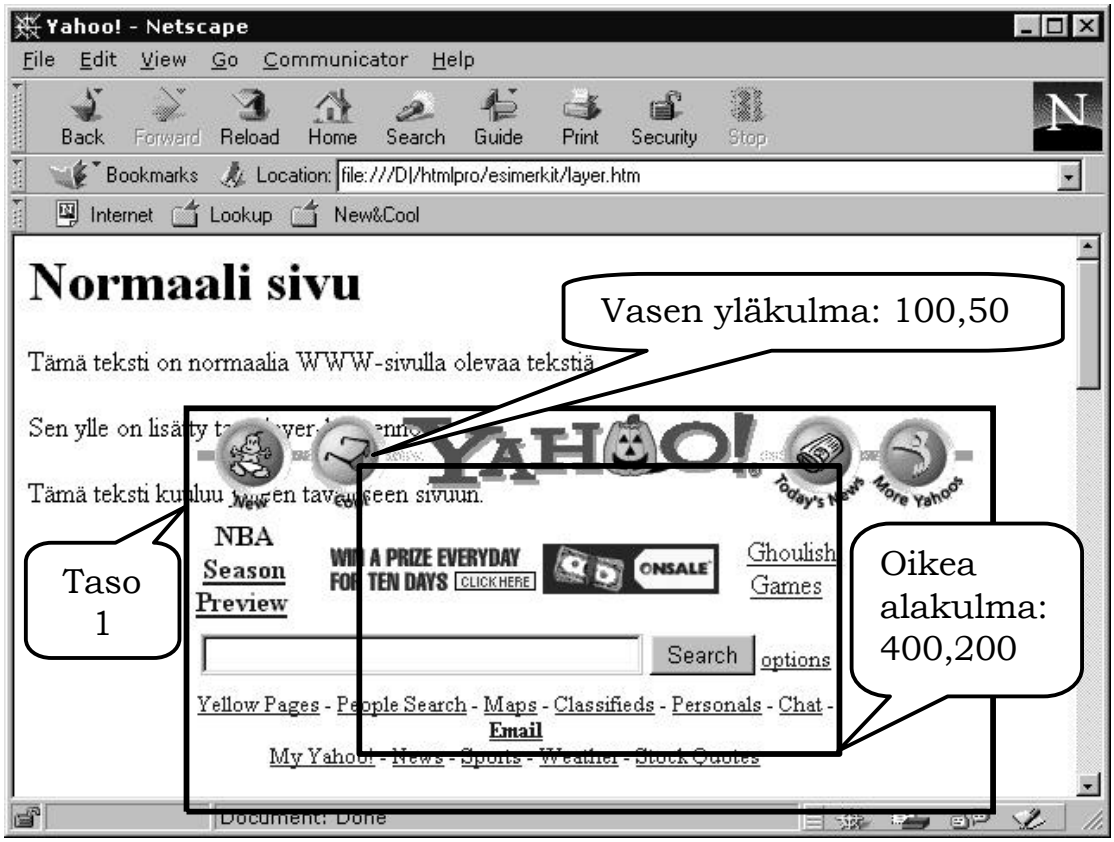
Leikattavat alueet määritellään antamalla leikattavan alueen kulmat. Edellisessä esimerkissä tasossa näkyvää **Yahoon** kotisivua voidaan muokata siten, että siitä näkyy vain osa:

```
<h1>Normaali sivu</h1>
<p>Tämä teksti on normaalia WWW-sivulla olevaa
tekstiä.
<p>Sen ylle on lisätty taso layer-komennolla.

<layer top=100 left=100 width=300 height=200
```

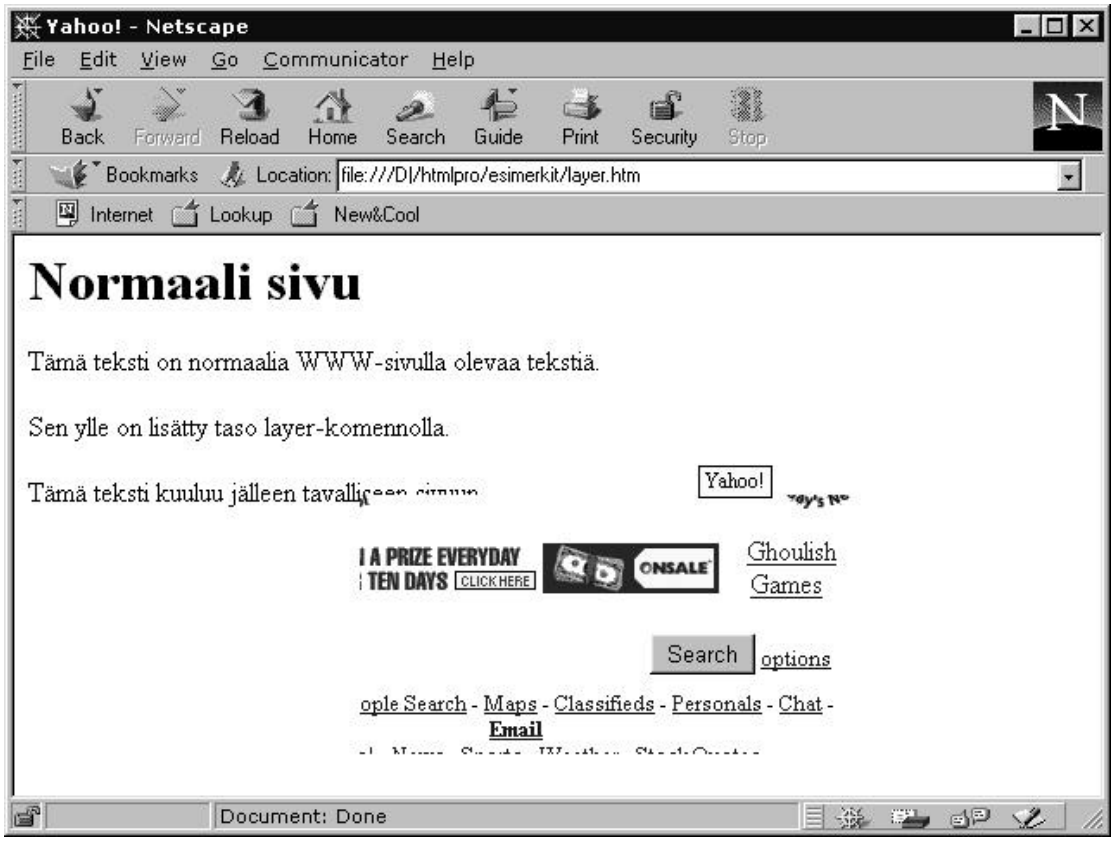
```
src="http://www.yahoo.com" bgcolor="White"
clip="100,50,400,200" id=taso1>
</layer>

<p>Tämä teksti kuuluu jälleen tavalliseen sivuun.
```

Attribuutti **clip** määrittelee, että **Yahoo**n kotisivusta näytetään vain osa, jonka vasen yläkulma on 100 pikseliä tason vasemmasta reunasta ja 50 pikseliä yläreunasta ja jonka oikea alakulma on 400 pikseliä tason reunasta ja 200 pikseliä yläreunasta.

Sivu ennen leikkaamista:



Kuvaruudulla näet, miltä sivu näyttää, kun tasossa on käytetty **clip**-attribuuttia:



Clip-attribuutissa voidaan antaa myös vain kaksi arvoa. Nämä kertovat silloin leikattavan kohdan oikean alakulman - vasemmaksi yläkulmaksi oletetaan tason vasen ylänurkka.

Kun sijoitat sivulle useita tasoja ja ne menevät päällekkäin, viimeksi lisätty taso peittää edellisen. Voit kuitenkin määritellä tasolle järjestyksen, jossa ne näkyvät. Tämä tapahtuu antamalla **layer**-komennoille jokin seuraavista attribuuteista: **z-order**, **above** tai **below**.

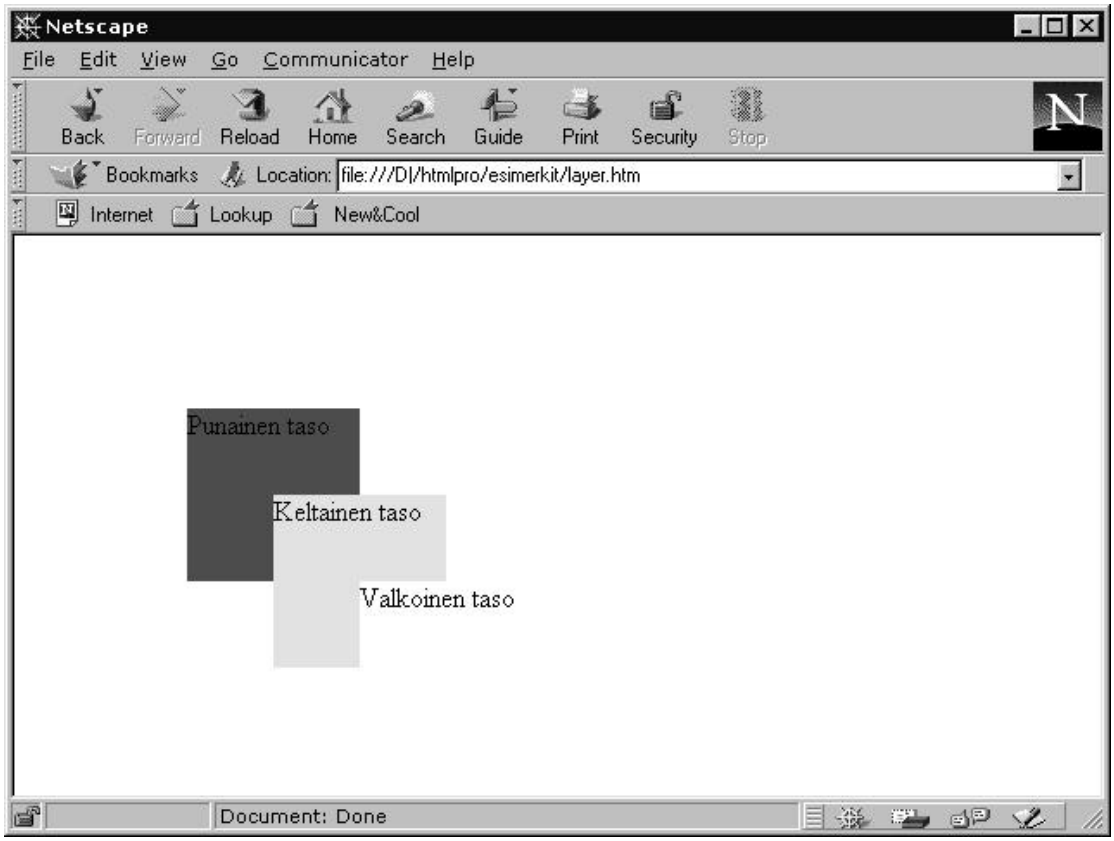
Attribuuteilla **above** ja **below** voit määritellä tasojen päällekkäisyysjärjestyksen. Kun olet antanut kaikille tasoille nimen **id**-attribuutilla, voit kertoa niiden avulla järjestyksen.

Otetaan esimerkki, jossa on kolme tasoa nimiltään "punainen", "keltainen" ja "valkoinen". Kaikki tasot ovat samankokoisia mutta hieman eri kohdissa niin, että ne näkyvät toistensa alta:

```
<layer id=punainen bgcolor="Red" top=100 left=100
width=100 height=100>
<p>Punainen taso
</layer>
```

```
<layer id=keltainen bgcolor="Yellow" top=150 left=150
width=100 height=100>
<p>Keltainen taso
</layer>

<layer id=valkoinen bgcolor="White" top=200 left=200
width=100 height=100>
<p>Valkoinen taso
</layer>
```

Tasot tulevat selaimessa näkyviin määrittelyjärjestyksessä: punainen alimmaisena, sitten keltainen ja päällimmäisenä valkoinen:



Voit lisätä tasojen **layer**-komentoihin attribuuteilla **above** ja **below**, missä järjestyksessä haluat tasojen näkyvän.

Esimerkiksi keltaisen tason **layer**-komento voisi näyttää tältä:

```
<layer id=keltainen bgcolor="Yellow" top=150 left=150
width=100 height=100 above="punainen">
```

Tämä kertoo, että keltainen taso luodaan siten, että punainen taso on järjestyksessä sitä ylempänä - punainen taso peittää keltaisen tason.

Voit määritellä tasoille myös syvyysjärjestyksen. Tämä tapahtuu attribuutilla **z-order**. Voit antaa tasoille numeron, joka kertoo niiden syvyyden. Päällimmäiseksi selaimen ruudulle päätyy aina se taso, jonka z-arvo on suurempi. Jos punaiselle tasolle on määritelty z-arvoksi 1, keltaiselle 2 ja valkoiselle 3, valkoinen taso tulee näkyviin ylimmäisenä, sitten keltainen taso ja alimmaisena punainen taso. Arvot voivat olla positiivisia numeroita.

Tasosta voidaan tehdä piilotettu attribuutilla **visibility**. Kun attribuutille annetaan arvo **hide**, taso ei näy ollenkaan kuvaruudulla. Attribuutilla **show** taso saadaan näkyväksi. Kolmas attribuutti, **inherit**, kertoo että tason näkyvyys on sama kuin tasolla, jonka sisään se on tehty. Piilotetuista tasoista on hyötyä silloin kun teet interaktiivisia sivuja: voit kytkeä JavaScript-ohjelmalla tason näkyviin tai pois näkyvistä halutun logiikan mukaisesti.

Tasoja osaavat näyttää vain uusimmat selainmallit (Netscape ja Internet Explorer versio 4). Sellaisille selaimille, jotka eivät osaa näyttää tasoja, voi määritellä vaihtoehtoisen tekstin komennolla **nolayer**. Aloitus- ja lopetuskomennon välissä oleva teksti tulee näkyviin vanhemmissa selainmerkeissä.

```
<nolayer>
Valitettavasti selaimesi ei osaa näyttää tasoja. Voit
katsella tasottomia versioita sivuistamme!
</nolayer>
```

Layer-komennoilla tehdyt tasot peittävät sivulla olevat tavalliset tekstit. Jos haluat tehdä tasoja, jotka sijoittuvat sivulla tekstin lomaan, voit käyttää **ilayer**-komentoa. Tässä komennossa käytetään samoja attribuutteja kuin **layer**-komennossa, mutta sille ei määritellä paikkaa vaan paikka määräytyy sen mukaan, mihin olet sijoittanut komennon.

```
<h1>Otsikko</h1>
<p>Tämän kappeleen sisään sijoitetaan taso.
<br><ilayer bgcolor="Yellow" width=100 height=100>
<p>Tämä on inflow layer.
</ilayer>
```

```
<br>
teksti jatkuu...
```

TASOT TYYLIMÄÄRITTELYIDEN AVULLA

Tasot voidaan määritellä myös käyttäen tyylimäärittelyitä. Kun tyylimäärittelyyn lisätään määrittely position, se saa aikaan tason. Määrittely **position:absolute** tekee samanlaisen tason kuin **layer**-komento, kun taas määrittely **position:relative** tekee **iframe**-tyyppisen tason.

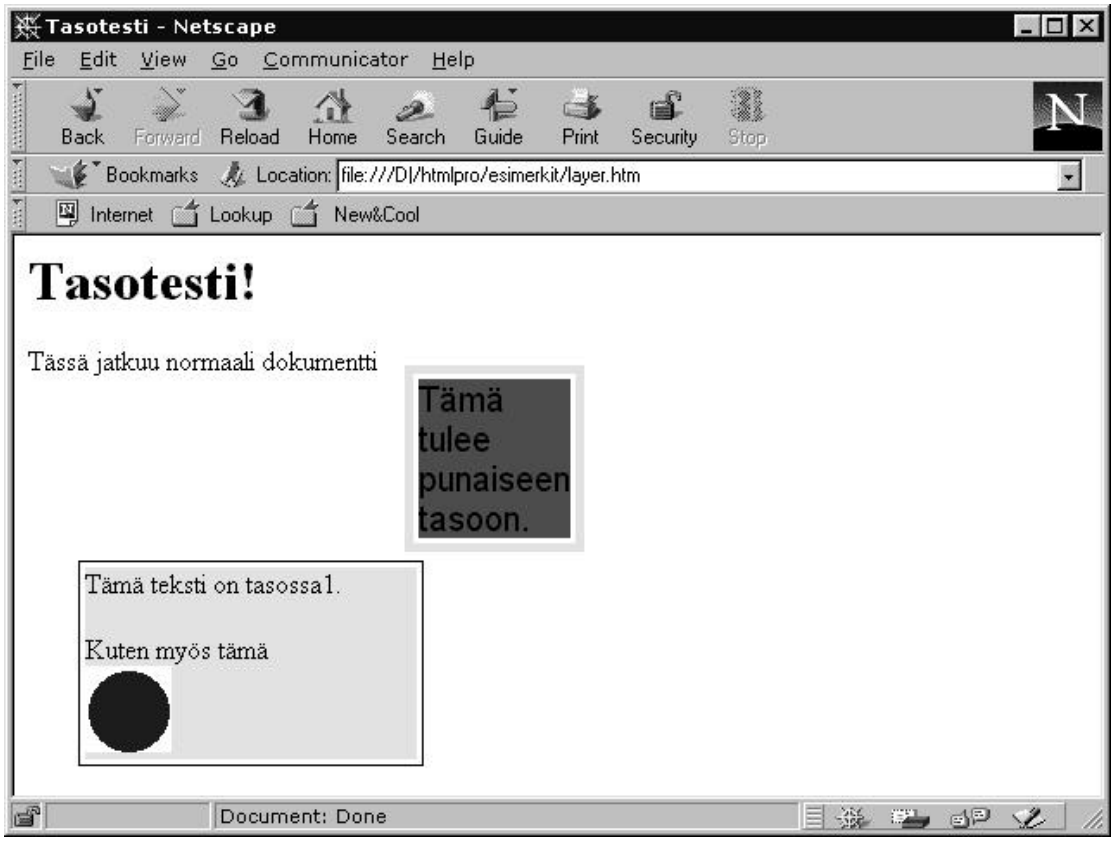
```
<h1>Otsikko</h1>
<p>Normaalia tekstiä.
<div style="position:absolute; background-
color:yellow; top:200px; left:100px">
<p>Tämä on layer!
</div>
<p>Tässä tulee taas normaalia tekstiä.
```

Voit määritellä tyylimäärittelyissä valmiiksi tasolle annettavat määrittelyt antamalla ennen määrittelyitä tyylille nimen ja käyttämällä tätä nimeä myöhemmin HTML-komennoissa attribuutissa **id**:

```
<html>
<head>
<title>Tasotesti</title>
<style type="text/css">
<!--
#keltainen {position: absolute;
top: 5cm; left: 1cm; background-color: Yellow;
border-width: 1; width: 200px;
}

#punainen {position: absolute;
top: 2cm; left: 6cm; background-color: Red;
width: 100px; font-size: 20px;
font-family: sans-serif;
border-width: 5; border-color: Yellow;
}
-->
</style>
</head>
```

```
<body>
<h1>Tasotesti!</h1>

<div id=keltainen>
<p>Tämä teksti on tasossa1.
<p>Kuten myös tämä
<br><img src="pallo.gif">
</div>

<p id=punainen>Tämä tulee punaiseen tasoon.</p>

<p>Tässä jatkuu normaali sivu
</body>
</html>
```

Näet selaimessa kuinka tyylimäärittelyt vaikuttavat tasojen ulkonäköön:



Voit käyttää tasojen tyylimäärittelyissä samoja komentoja kuin muuallakin. Tasoihin liittyy lisäksi omia määrittelyitä, joilla saadaan aikaan samoja asioita kuin **`layer`**-komennon attribuuteilla.

Tasolle annetaan nimi tyylitiedostossa sijoittamalla sen eteen risuaitamerkki (#). Määrittelyllä **`position`** kerrotaan onko kyseessä taso, jonka paikka

on määrätyssä kohdassa ruudulla vai siirtyykö sen sisältö HTML-tekstin mukana. Esimerkki:

```
#taso1 {position: absolute;}
/* tietyssä paikassa oleva taso */
#taso2 {position: relative;}
/* siirtyy HTML-tekstin mukana */
```

Tason paikka annetaan määrittelyillä **top** ja **left**. Ne voivat saada arvokseen yksiköitä pisteinä (**pt**), tuumina (**in**), senttimetreinä (**cm**), pikseleinä (**px**) tai prosenttikokona (**%**):

```
top: 200px; left: 5cm;
```

Tason koko määritellään määrittelyillä **width** ja **height**, jotka voivat saada samoja arvoja kuin paikan määrittelyt:

```
width: 100px; height: 10pt;
```

Jos haluat tuoda tason sisällön muusta tiedostosta tai osoitteesta, voit tehdä sen määrittelyllä **source-include**:

```
source-include:url("http://www.yahoo.com");
```

Tason leikkaus kerrotaan määrittelyllä **clip**, joka saa arvonsa kuten vastaava **layer**-komennon attribuutti:

```
clip:rect(0,100,100,0);
```

Tasojen päällekkäisyysjärjestyksen määrittelyyn on olemassa määrittely **z-index**, joka toimii samoin kuin **layer**-komennossa. Attribuutteja **above** ja **below** vastaavia määrittelyitä ei ole.

```
z-index: 5;
```

Tason näkyvyyttä hallitaan määrittelyllä **visibility**:

```
visibility: hide;
```

Tason taustaväriä ja taustakuvaa voidaan muuttaa määrittelyillä **background-color** ja **background-image**:

```
background-color: Green;
background-color: FFFFFF;
```

```
background-image:url("dots_grey.gif");
```

LISÄTIETOJA

Lisätietoja tasoista ja niiden käytöstä sekä erilaisia esimerkkejä löydät sivulta

```
http://developer.netscape.com/library/
documentation/communicator/dynhtml/index.htm
```

LAYOUT-TRIKKEJÄ

Kun haluat viilata WWW-sivujesi ulkoasua erittäin tarkasti, voit käyttää verkossa yleisesti käytössä olevia layout-trikkejä. Ota kuitenkin huomioon, että näiden käyttäminen saattaa tehdä sivuista oudosti näkyviä joissain vanhemmissa selainmerkeissä.

KUVIEN PALOITTELU JA YHDISTÄMINEN

Joskus tarvitset erikoisemman kokoisia kuvia. Saatat haluta, että kuvasi kiertävät tekstin tai että sama kuva siirtyy kehyksestä toiseen saumattomasti.

Jos haluat tehdä sivuillesi epäsäännöllisen kokoisia kuvia, kuvia jotka kiertävät tekstin, voit koota kuvasi useista paloista. Tästä on hyötyä myös silloin, jos haluat sijoittaa kuvan sisään vaikkapa animoidun kuvan.

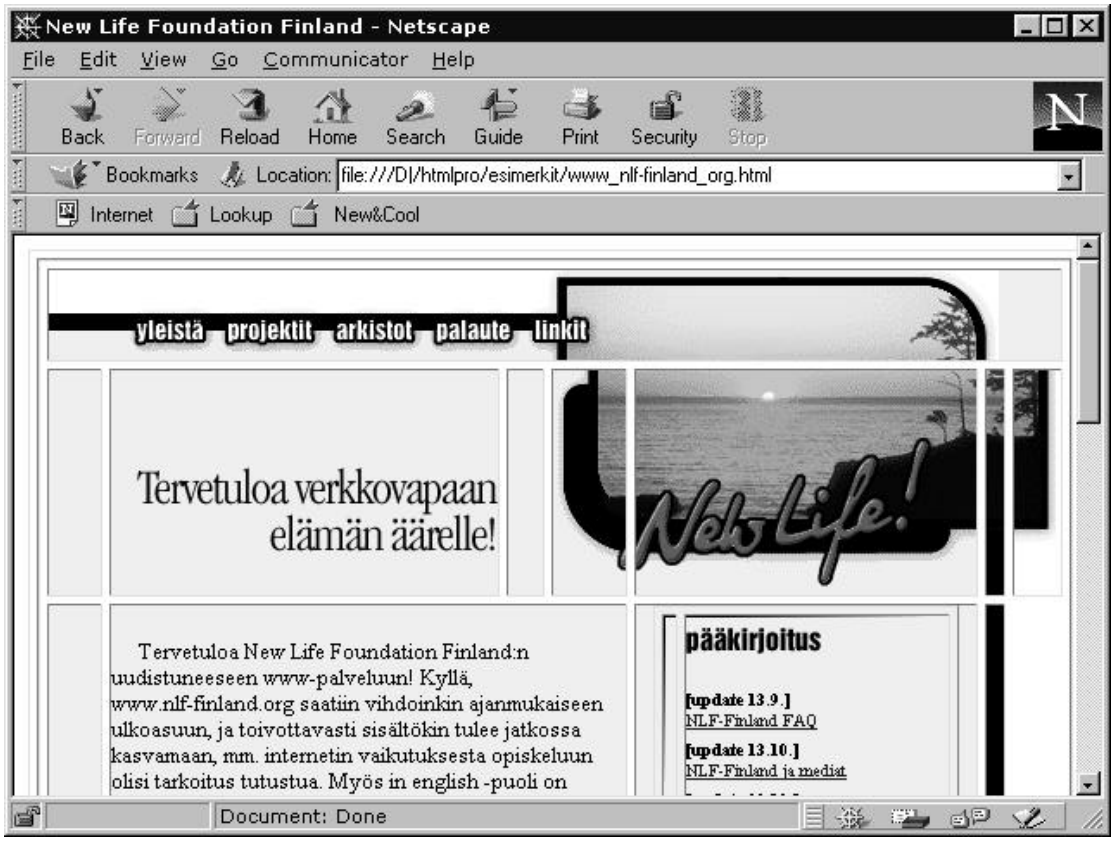
Himos-lomien etusivulla on iso kuva, jonka sisällä on **gif**-animaatio. Koko iso kuva olisi voitu toteuttaa animoituna **gif**-kuvana, mutta silloin siitä olisi tullut erittäin iso ja hitaasti ladattava. Sen sijaan sivulla oleva yksi iso kuva on koottu pienemmistä kuvista, jotka on sijoitettu tarkasti kiinni toisiinsa.

Jos sivulla oleva taulukko hajotetaan osiin ja kuvien väliin lisätään hieman tyhjää tilaa, näet kuinka kuva on tehty:

Ylhäällä näet kuinka kuva on koottu taulukkoon, josta on näkyvissä kaksi solua. Alempi solu sisältää kaksi kuvaa, joista oikeanpuoleinen on **gif**-animaatio.

Kun yhdistät kuvien osat, taulukot ja taulukoiden solujen taustavärit, voit toteuttaa monimutkaisiakin rakenteita. Esimerkkinä on **New Life Foundationin** sivu:

Kun sivu jaetaan osiin, huomaat että etusivun suuri kuva on koottu osista. Kuvan alle tulevien tekstipalstojen leveys saadaan määriteltyä kuvien osien avulla. Tekstiosuutta reunustavaa mustaa reunusta ei ole tehty ollenkaan **gif**-kuvilla vaan kyseessä on kapea solu, jonka taustaväri on määritelty mustaksi.



Kuvien sovitus taulukoihin siten, että ne sopivat täsmälleen toisiinsa, vaatii tarkkaa suunnittelua. Kun rakennat taulukkoa, jossa viereisten solujen yhteensopivuus on tärkeää, muista kytkeä taulukon reunat ja marginaalit pois päältä:

```
<table border=0 cellspacing=0 cellpadding=0>
```

Jos teet sivua, jonka kehykset sisältävät toisten kehyksien suhteen suunniteltuja kuvia, kytke kehyksistä reunat pois komennoilla

```
<frameset border=0 frameborder=0  framespacing=0
framepadding=0>
```

... ja

```
<frame marginwidth=0 marginheight=0>
```

PIKSELIKUVAT

Kuten muistat, kuvien kokoa voitiin säätää antamalla kuvakomennolle attribuutit **width** ja **height**. Tällä tavalla saatoit tehdä pienestäkin kuvasta suuren venyttämällä sitä:

```
<img src="pieni.gif" width=200 height=200>
```

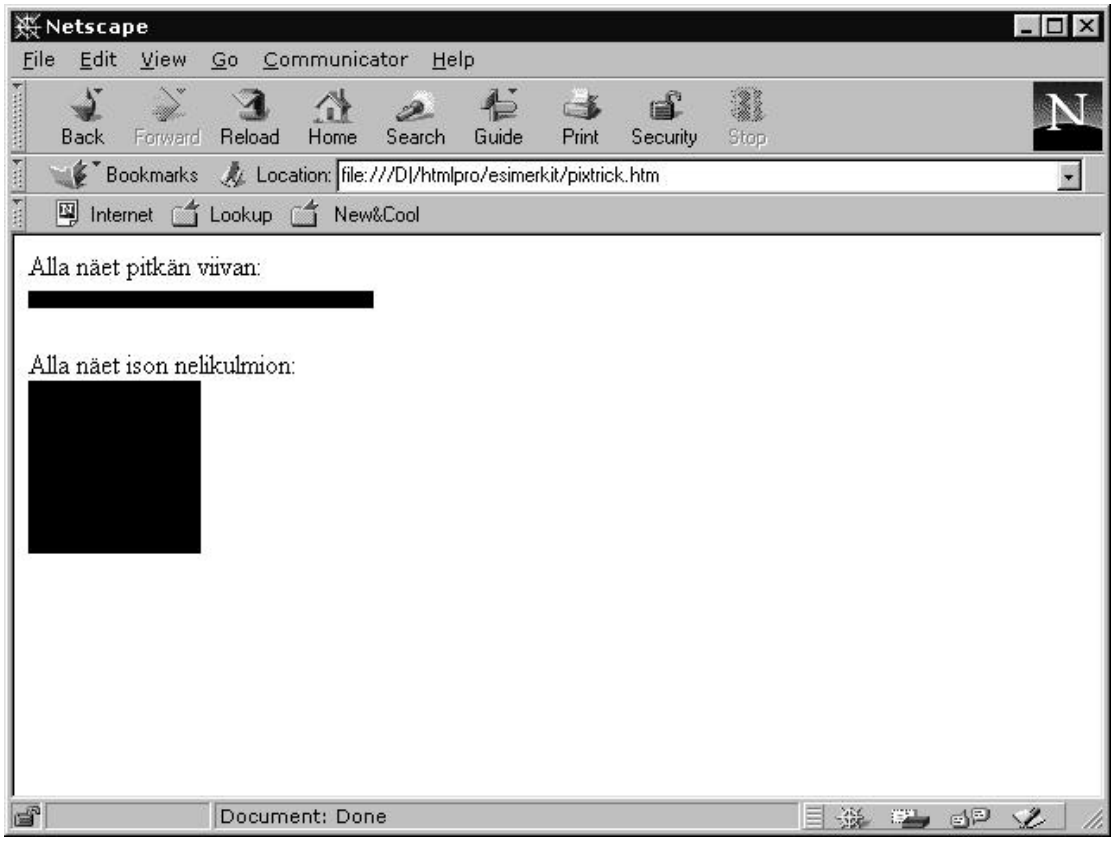
Tätä ominaisuutta voi käyttää hyväkseen silloin, kun haluaa tehdä sivuja, joissa haluaa määritellä elementtien paikkoja pikselin tarkkuudella.

Jos teet kuvan, joka on vain yhden pikselin kokoinen, voit venyttää sitä minkä kokoiseksi tahansa. Tee **Paint Shop Pro**lla kuva, jonka koko on 1x1 ja joka koostuu yhdestä mustasta pisteestä. Anna tiedostolle nimeksi **musta.gif**.

Tämän kuvan avulla voit piirtää viivoja ja laatikoita venyttämällä pikseliä sopivan kokoiseksi:

```
<p>Alla näet pitkän viivan:
<br>
<img src="musta.gif" width=200 height=10>
<p>Alla näet ison nelikulmion:
<br>
<img src="musta.gif" width=100 height=100>
```

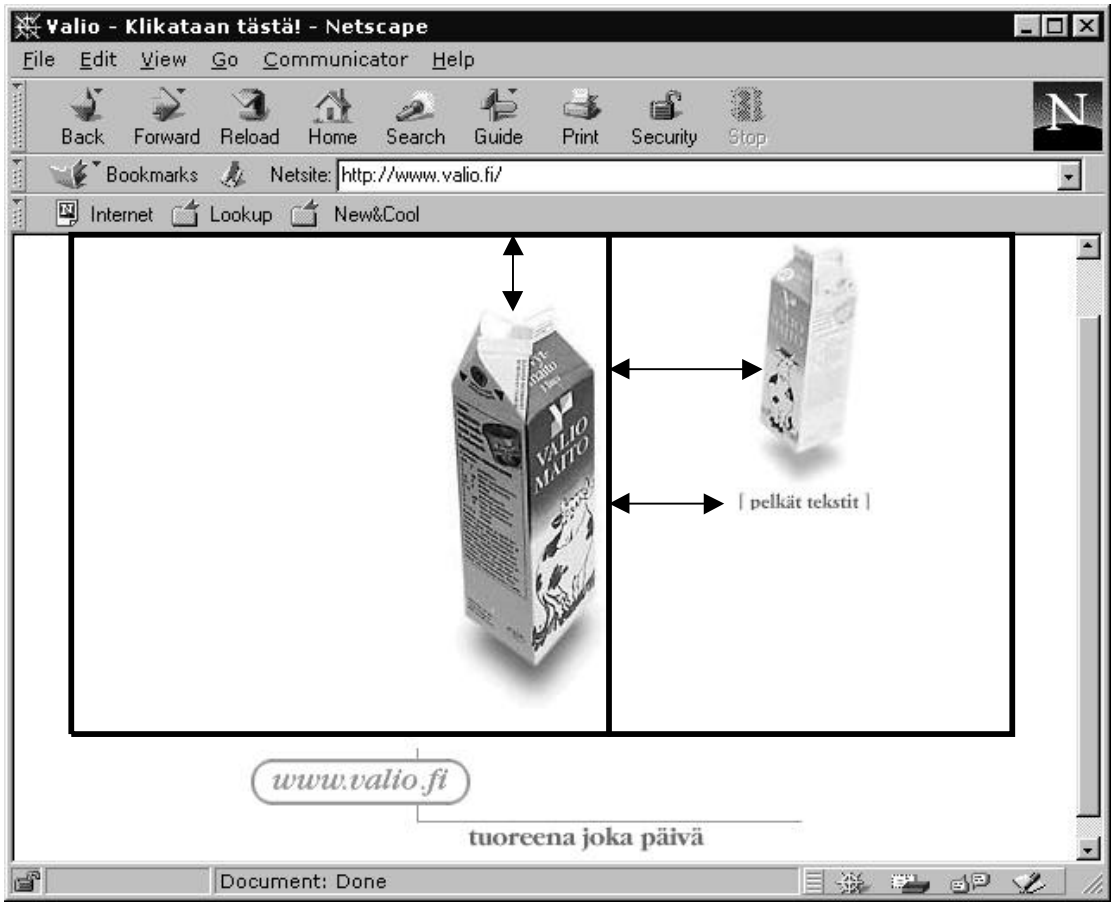
Selaimessa näet, kuinka piste venyy antamiesi määrittelyiden mukaan:



Tällä tavalla voit tehdä myös kuvioita, joiden koko on suhteellinen selaimen ikkunaan nähden:

```
<img src="musta.gif" width="100%" height=20>
```

Jos teet yhden pikselin kokoisen kuvan, jonka väri on läpinäkyvä, voit tehdä läpinäkymättömiä elementtejä. Niiden avulla voit pakottaa taulukon soluja tietyn levyisiksi, tai tehdä tyhjää tilaa haluamiisi kohtiin näytöllä.



Esimerkiksi Valion sivuilla kuvaruudulla näkyvät maitopurkit on sijoitettu ruudulle käyttämällä kaksisoluista taulukkoa (kuvassa katkoviiva) ja solujen sisällä oikeille paikoilleen käyttämällä läpinäkyviä yhden pikselin kokoista **gif**-kuvia jotka on venytetty halutun kokoisiksi (kuvassa merkitty nuolilla).

Voit käyttää läpinäkyvän pikselin kanssa myös kuvien attribuutteja **vspace** ja **hspace**. Nämä eivät venytä kuvaa vaan tekevät sen molemmille puolille halutun määrän tyhjää tilaa:

```
<img src="lapinakyva.gif" vspace=20 hspace=0>
```

Yllä oleva esimerkki tekee selaimen näytölle pystysuoran 41 pikselin kokoisen tyhjän tilan (kuvan ylä- ja alapuolelle lisätään 20 pikseliä tyhjää tilaa).

Näiden attribuuttien etuna on se, että vaikka käyttäjä ei lataisikaan kuvia, niiden tilalle tuleva lataamattoman kuvan ikoni ei tule näkyviin ruudulle. Kuvia lataamaton käyttäjä saa näytölleen korkeintaan yhden pikselin kokoisen merkin.

Kun teet tyhjää tilaa pikselikuvilla, älä unohda käyttää kuvakomennoissa attribuuttia **`alt=""`**. Tällä tavalla tilan tekemiseen tarkoitetut kuvat eivät tule näkyviin niissä selaimissa, jotka eivät osaa näyttää kuvia.

Pikselikuvien käyttöä elegantimpi tapa sijoitella kuvia ja tekstiä tietokoneen ruudulle on tasojen käyttö. Tasot eivät kuitenkaan toimi vanhemmissa selainversioissa. Tulevaisuudessa pikselikuvien käyttäminen kuitenkin vähenee samalla kun uudet selainmallit muuttuvat yhä suosituimmiksi.

MULTIMEDIA SIVUILLA

WWW-sivuille on mahdollista lisätä kuvien lisäksi myös monia multimediakomponentteja. Monilla sivuilla on käytössä esimerkiksi äänitehosteita, animaatioita ja kokonaisia multimediasovelluksia.

ÄÄNI

Äänien lisääminen WWW-sivulle tapahtuu yksinkertaisimmillaan tekemällä linkki äänitiedostoon. Jos sinulla on musiikinpätkä tiedostossa nimeltään **musiikki.wav**, voit tehdä siihen linkin vaikkapa seuraavalla tavalla:

```
<a href="musiikki.wav">Klikkaa tästä niin kuulet
musiikkia</a>
```

Voit liittää äänitiedoston sivuille myös komennolla **bgsound**. Tälle komennolle annetaan attribuutit **src** ja **loop**. Ensimmäinen attribuutti kertoo, mistä äänitiedosto löytyy - se voi olla joko tiedostonnimi tai URL-osoite, aivan kuten linkeissäkin. Jälkimmäinen parametri kertoo, montako kertaa ääni toistetaan. Esimerkki:

```
<bgsound src="canyon.mid" loop=2>
<!-- toistaa kappaleen kaksi kertaa-->
```

Bgsound-komento toimii tällä hetkellä vain Internet Explorer -selaimessa. Jos haluat liittää äänitiedostoja siten, että kaikki lukijasi voivat kuulla ne, voit käyttää **embed**-komentoa.

Yllä oleva esimerkki **embed**-komennolla toteutettuna näyttää seuraavalta:

```
<embed src="canyon.mid" width=144 height=60>
```

Myös **embed**-komennolle kerrotaan attribuutilla **src**, mistä äänitiedosto löytyy. **Embed**-komennolla lisätty äänitiedosto tuo selaimen näytölle pienen ohjauspaneelin, josta käyttäjä voi käynnistää ja sammuttaa äänen soitta-

misen. Attribuutilla **loop** voit määritellä, miten monta kertaa ääni toistetaan.

Ohjauspaneelin kokoa määritellään attribuuteilla **width** ja **height**, aivan kuten kuvienkin kokoa. Paneelille voi antaa myös attribuutin **align**, joka toimii samalla tavalla kuin kuvia sijoitettaessa.

Normaalisti käyttäjä voi itse käynnistää ja pysäyttää äänen toistamisen ohjauspaneelissa näkyvissä olevilla painikkeilla. Voit määritellä äänen käynnistymään automaattisesti lisäämällä **embed**-komentoon attribuutin **autostart=true**.

Ohjauspaneelin voi kätkeä näkyvistä attribuutilla **hidden=true**. Jos käynnistät musiikin soittamisen automaattisesti, käyttäjälle ei jää kuitenkaan mahdollisuutta pysäyttää musiikin soittamista, joten käytä tätä ominaisuutta harkiten!

Voit käyttää sivuillasi ääninä **wav**-, **aiff**- ja **au**-muodossa tallennettuja ääninäytteitä. Windows-ohjelmilla tallennetut ääninäytteet ovat yleensä **wav**-muodossa.

Lisäksi voit liittää sivuille Midi-muodossa olevia kappaleita, joiden tarkennin on **mid**. Midi-tiedostot sisältävät varsinaisen äänen sijasta nuotteja ja muita ääntä kuvaavia parametrejä, joten ne ovat paljon pienikokoisempia kuin varsinaiset ääninäytteet. Midi-tiedostot saattavat kuitenkin soida eri tavalla eri lukijoilla, riippuen heidän käyttämästään äänikortista. Jos haluat liittää sivuillesi musiikkia, käytä mieluiten Midi-tiedostoja, koska ne ovat paljon pienikokoisempia kuin muissa muodoissa olevat ääninäytteet.

Jos haluat kokeilla ääninäytteitä ja sinulla on käytössäsi Windows 95, löydät runsaasti äänitiedostoja omalta tietokoneeltasi C-levyn hakemistosta **Windows/Media**.

VIDEOPÄTKÄT

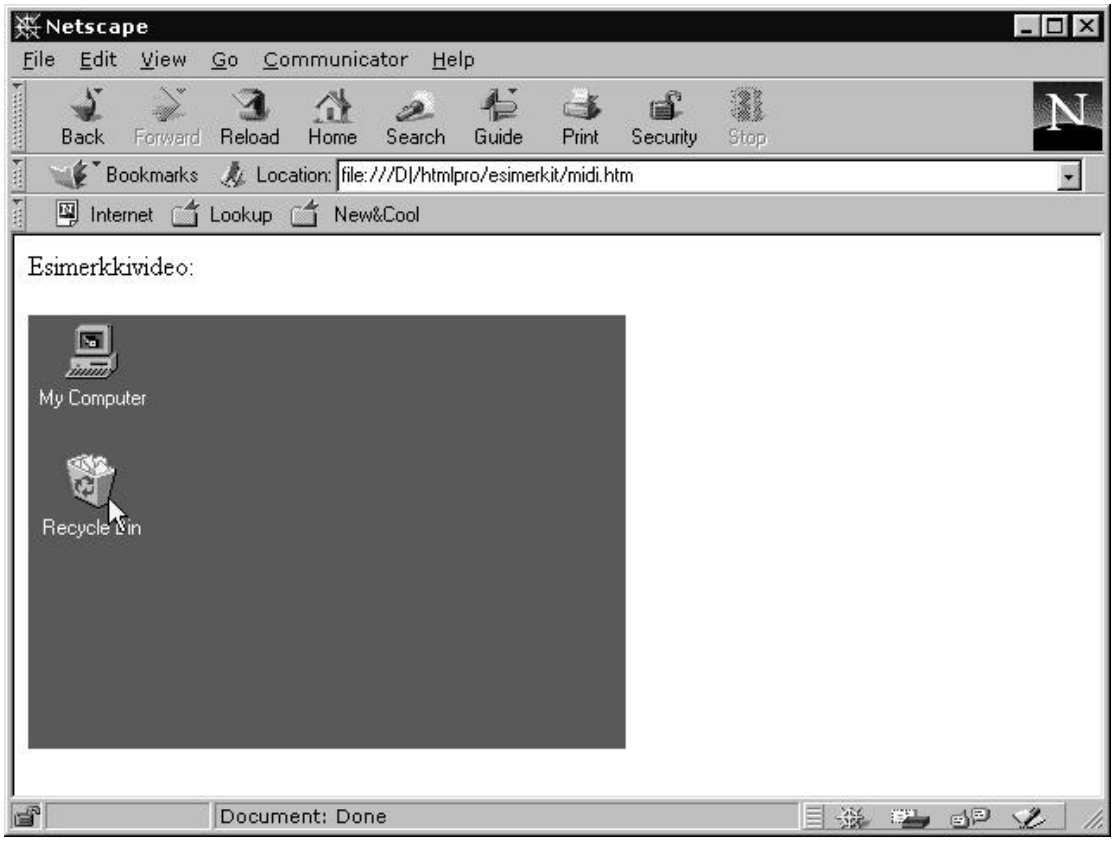
Videotiedostojen lisääminen sivuille tapahtuu samalla periaatteella kuin kuvienkin lisääminen. Voit tehdä linkin suoraan siihen tiedostoon, jossa video on:

```
<a href="dragdrop.avi">Klikkaa tästä niin näet
videon!</a>
```

Voit myös upottaa videopätkän suoraan sivulle käyttämällä **embed**-komentoa aivan kuten äänitiedostoa liitettäessä. Voit käyttää lisäämisessä samoja attribuutteja kuin äänien tapauksessakin. Esimerkki:

```
<p>Esimerkkivideo:
<p><embed src="dragdrop.avi" width=346 height=251
autostart=true>
```

Jos käytät Windows 95:ttä, löydät animaation **dragdrop.avi** C-levyltäsi hakemistosta **Windows/Help** tai Windows 95 -asennuslevyltä.



Voit lisätä sivuillesi videotiedostoja useissa eri muodoissa. Yleisin on Windowsin oma animaatiomuoto, **AVI**, mutta muita useilla WWW-sivuilla nähtyjä formaatteja ovat myös **Quicktime** ja **Vxtreme**.

Käyttäjät eivät voi katsoa kaikkia formaatteja sellaisinaan: esimerkiksi **Quicktime**-animaatioiden katsomiseksi selaimeen täytyy asentaa Applen toimittama **plug-in**. **Plug-init** ovat selaimen lisäosia, jotka tarjoavat mah-

dollisuuden katsoa sellaisia **embed**-komennoilla upotettuja formaatteja, joiden katsominen ei normaalisti ole mahdollista.

Kun käyttäjä katselee sivua, jolle on upotettu **plug-iniä** vaativa animaationpätkä eikä hänellä ole käytössään vaadittua lisäosaa, selain voi ehdottaa hänelle sen hankkimista ja asentamista. Tämä toteutetaan lisäämällä **embed**-komentoon attribuutti **pluginspace**, johon lisätään osoite, josta **plug-inin** voi hankkia.

Quicktime-**plug-inin** voi kopioida osoitteesta
http://quicktime.apple.com. Tämä osoite lisätään **embed**-komentoon seuraavasti:

```
<embed src="elokuva.qt"
pluginspace="http://quicktime.apple.com/">
```

Hyviin tapoihin kuuluu myös mainita sivulla siitä, että animaation katselu vaatii tietyn **plug-inin** ja antaa linkki paikkaan, josta käyttäjä voi halutessaan kopioida sen itselleen.

Videotiedostot ovat erittäin isokokoisia - jo muutaman sekunnin mittaisen animaation siirtäminen modeemin yli kestää useita minuutteja. Harkitse sen vuoksi tarkasti, onko videoiden liittäminen sivuillesi tarpeellista. Tarjoa käyttäjälle aina vaihtoehto saada samat tiedot muillakin tavoilla - tarjoa käyttäjälle esimerkiksi valikoima videosta otettuja kuvia.

YHDISTETTY MULTIMEDIA

Yhdistetty multimedia sisältää kuvia, ääntä, animaatiota ja erilaisia interaktiivisia osia. Tällaisia osia voit toteuttaa esimerkiksi **Macromedian Shockwavella** ja **Flashilla**.

Näiden avulla voit ohjelmoida WWW-sivun sisään alueita, joissa voit näyttää kuvia, soittaa ääniä ja toteuttaa vaikkapa interaktiivisen pelin.

Multimediakomponentit lisätään sivun sisään **embed**-komennolla aivan kuten videot ja äänetkin. Lisätietoja **Shockwavesta** ja **Flashista** saat **Macromedian** kotisivulta osoitteesta

```
http://www.macromedia.com
```

IMUROITAVAT OHJELMAT

Voit tarjota kotisivullasi tietokoneohjelmia tai erilaisia tiedostoja käyttäjän imuroitavaksi. Jos sinulla on ajettava ohjelma, jota haluat jakaa sivullesi saapuville käyttäjille, tee siihen suora linkki:

```
<a href="ohjelma.exe">Imuroi ohjelma klikkaamalla
tästä</a>
```

Useimmat palvelimet tunnistavat ajettavien ohjelmien nimen päätteestä (**.exe**), että kyseessä on ajettava ohjelma ja välittävät tämän tiedon lukijan selaimelle. Lukijan selain antaa käyttäjälle vaihtoehdon joko ajaa ohjelma suoraan tai tallentaa se levylle.

Voit myös pakata ohjelman tai tiedostot, joita haluat jakaa. Kun pakkaat tiedostot yleisimmillä pakkausohjelmilla ja sijoitat pakatun tiedoston hakemistoosi, voit niin ikään tarjota sen imuroitavaksi tekemällä linkin tiedostoon:

```
<a href="ohjelma.zip">Imuroi ohjelma klikkaamalla
tästä</a>
```

Hyviin tapoihin kuuluu ilmoittaa linkin yhteydessä imuroitavan tiedoston koko. Tällä tavalla käyttäjä pystyy arvioimaan, kauanko tiedoston siirtoon tulee kulumaan aikaa.

Jos ajettavien ohjelmien siirto ei tunnu onnistuvan, varmista operaattoriltasi, että WWW-palvelin tunnistaa tiedostonnimien tarkentimet oikein ja antaa selaimelle niitä vastaavan tiedon.

JAVASCRIPT

JavaScript on tapa sijoittaa pieniä ohjelmia WWW-sivujen sisään. Nämä ohjelmat voivat tulostaa käyttäjän ruudulle erilaisia tekstejä tai muokata eri tavoilla WWW-sivun tai selaimen ominaisuuksia.

JAVASCRIPT-PERUSTEET

JavaScript-komennot sijoitetaan sivun sisään. Kun selain lataa sivua käyttäjän näytölle, se suorittaa vastaantulevat komennot.

Komennot sijoitetaan **script**-komentojen väliin. Alla näet esimerkin JavaScript-ohjelmasta, joka tulostaa käyttäjän ruudulle tekstiä. Esimerkissä JavaScript-komennot on sijoitettu tiedostoon **head**-komentojen väliin. Silloin selain ajaa komennot heti sivulle saavuttuaan.

```
<html>
<head><title>Javascriptiä</title>
<script language=”JavaScript”>
<!--
document.write(”<h1>Heippa, maailma!</h1>”);
// -->
</script>
<body>
<p>Tästä alkaa tavallinen HTML.
</body> </html>
```

Komennot on sijoitettu HTML-kommenttimerkkien väliin siksi, että ne eivät näkyisi selaimissa, jotka eivät tunne JavaScriptiä.

Script-komennossa määritellään attribuutilla **language** käytetyn kielen tyyppi. Tämän avulla voit määritellä esimerkiksi käyttämäsi JavaScript-version numeron.

JavaScript-osuudessa käytetty komento **document.write** tulostaa HTML-tiedoston sisään sulkujen ja lainausmerkkien välissä olevan tekstin. Kun katselet esimerkkiä selaimessa, huomaat kuinka ohjelma tulostaa lukijan ruudulle HTML-tekstiä:

Komennot suoritetaan siinä järjestyksessä kuin selain ne käsittelee, eli sivun alusta loppuun päin:

```
<html>
<head><title>Javascriptiä</title>
<script language=”JavaScript”>
<!--
document.write(”<h1>Heippa, maailma!</h1>”);
// -->
</script>
<body>
<p>Tästä alkaa tavallinen HTML.
<script language=”JavaScript”>
<!--
document.write(”<p>Tämä tulostuu loppuun”);
// -->
</script>
</body> </html>
```

JavaScript-komentojen sisään voidaan sijoittaa kommenttimerkkejä, joilla on sama merkitys kuin HTML-kommenteillakin: ne näkyvät vain niille, jot-

ka katselevat itse koodia. Tällä tavalla voit tehdä koodin sisään omia muistiinpanojasi tai ottaa väliaikaisesti jonkin komennon pois.

Kommentit merkitään joko lisäämällä rivin alkuun kaksi kauttaviivamerkkiä (**//**) tai aloittamalla kommentoitava pätkä merkeillä **/*** ja lopettamalla se merkkeihin ***/**. Jälkimmäisellä tavalla voit tehdä useamman rivin mittaisia kommentteja.

```
// tämä tulostaa tekstiä
document.write("Heippa!");
```

Jokainen JavaScript-komento kirjoitetaan yleensä omalle rivilleen ja rivit lopetetaan aina puolipisteeseen.

```
document.write("<p>JavaScript on kivaa!");
document.write("<p>Vai mitä?");
```

JavaScriptissä voi käyttää lainausmerkkien tilalla tarvittaessa myös heittomerkkejä:

```
document.write('<p>Toimii myös!');
```

Tästä on hyötyä silloin, jos JavaScript-komentoja on sijoitettu HTML-komentojen sisään:

```
<a href="linkki.html"
onClick="document.write('klik!');">
```

Jos JavaScript-komennoissa on virhe, selain ilmoittaa siitä välittömästi:

```
<html>
<head><title>Javascriptiä</title>
<script language=”JavaScript”>
<!--
document.write('<h1>Heippa, maailma!</h1>”);
// -->
</script>
<body>
```

```
<p>Tästä alkaa tavallinen HTML.
</body> </html>
```

Esimerkin komennossa lainausmerkki oli vaihtunut heittomerkiksi. Selain antaa virheestä varoituksen sivua ladattaessa ja näyttää virheellisen kohdan:

MUUTTUJAT

JavaScript-ohjelmat voivat tallentaa tietoa muuttujiin ja käyttää tätä tietoa hyväkseen erilaisissa laskutoimituksissa ja operaatioissa. Muuttujia voi olla kolmea eri tyyppiä: numeromuuttujia, tekstimuuttujia ja Boolen arvoja (tosi tai epätosi). Lisäksi muuttujalle voidaan antaa arvo **`null`**, eli tyhjä.

Kun muuttujia käytetään ensimmäistä kertaa, niille annetaan jokin arvo. Muuttujan esittely tapahtuu komennolla **`var`**:

```
var i=1;
```

Tämän jälkeen muuttuja nimeltään **`i`** sisältää numeron 1. Voit käyttää muuttujaa esimerkiksi tulostuskomennossa:

```
document.write(i);
```

Esimerkki tulostaa selaimeen numeron 1. Muuttujia voidaan muokata vapaasti sen jälkeen, kun ne on alustettu:

```
var i=1;
document.write(i);
i=2;
document.write(i);
```

Yllä oleva esimerkki tulostaa ensin numeron 1 ja sitten numeron 2. Voit tehdä muuttujilla erilaisia laskutoimituksia ja tulostaa niiden arvoja:

```
var maito=3.50;
var suklaavanukas=4.95;
document.write("Ostokset maksoivat ");
document.write(maito+suklaavanukas);
document.write(" mk");
```

Voit tehdä muuttujilla peruslaskutoimituksia: + (ynnäys), - (vähennys), * (kertolasku), / (jakolasku), % (jakojäännös):

```
var tyopaiva=7.5;
var viikkotunnit=5*tyopaiva;
document.write("<p>Teet viikossa tunteja ");
document.write(viikkotunnit);
var kkpalkka=10000;
var viikkopalkka=kkpalkka/4;
document.write("<p>Viikkopalkkasi on ");
document.write(viikkopalkka);
document.write(" mk");
document.write(" ja tuntipalkkasi ");
document.write(viikkopalkka/viikkotunnit);
document.write(" mk.");
```

Pikalaskutoimituksilla voit muuttaa näppärästi yhden muuttujan arvoa:

Lisää yksi	**++**
Vähennä yksi	**--**
Lisää luku	**+=**
Vähennä luku	**-=**
Kerro luku	***=**
Jaa luku	**/=**

Voit siis lisätä muuttujan arvoa yhdellä seuraavalla näin:

```
var a=1;
a++;
// a on nyt 2
```

Alla oleva esimerkki lisää lukuun numeron 8:

```
var a=1;
a+=8;
// a on nyt 9
```

Muuttujiin voi sijoittaa myös merkkijonoja. Merkkijonoja määriteltäessä teksti ympäröidään lainaus- tai heittomerkeillä:

```
var nimi="Sami";
```

Muuttujia voidaan yhdistellä plusmerkillä. Yhdistelyä voi käyttää myös komentojen sisällä:

```
var etunimi="Bill";
var sukunimi="Clinton";
var kokonimi=etunimi+" "+sukunimi;
// etu- ja sukunimen väliin tulee
// välilyönti
document.write("Nimesi on "+kokonimi);
```

Esimerkki tulostaa ruudulle:

```
Nimesi on Bill Clinton
```

Voit lisätä merkkijonoihin erikoismerkkejä käyttämällä merkkikoodeja:

tabulaattori	**\t**
backspace	**\b**
form feed	**\f**
newline	**\n**
return	**\r**
lainausmerkki	**\"**

Lainausmerkkejä ei voi normaalisti sijoittaa tulostuskomentoon, koska ne tulkittaisiin tulostettavan tekstin lopuksi. Niiden sijasta voit käyttää lainausmerkin merkkikoodia:

```
var etunimi="William";
var sukunimi="Clinton";
var lempinimi="Bill"
var kokonimi=etunimi+" \""+lempinimi+
"\" "+sukunimi;
// etu- ja sukunimen väliin tulee
// välilyönti
document.write("Nimesi on "+kokonimi);
```

Esimerkki tulostaa ruudulle:

```
Nimesi on William "Bill" Clinton
```

Boolen arvot ovat muuttujia, jotka voivat saada arvon **true** tai **false**. Näitä arvoja voidaan käyttää esimerkiksi **if**-komennon testeissä (katso alempaa).

Muuttujia voidaan käsitellä myös bittioperaattoreilla: **&** (**AND**), **|** (**OR**), **^** (**XOR**), **~** (**NOT**), **<<** (**left shift**), **>>** (**sign-propagating right shift**) ja **>>>** (**zero-fill right shift**).

JavaScript-muuttujia on mahdollista käyttää myös HTML-komentojen attribuuteissa. Voit käyttää silloin JavaScript-entiteettejä. JavaScript-entiteetit näyttävät tavallisilta entiteeteiltä, mutta niissä sijoitetaan entiteetin nimen sijasta muuttujan nimi kaarisulkujen sisään, esimerkiksi **&{muuttuja};**. Alla olevasta esimerkistä näet, miten voit määritellä ohjelmallisesti väliviivan leveyden:

```
<script language="JavaScript">
var leveys=150;
</script>
<hr width="&{leveys};">
```

Esimerkki tekee selaimen ruudulle 150 pikseliä leveän väliviivan. JavaScript-entiteettejä voi käyttää vain komennon attribuuttien sisällä.

KONTROLLIKOMENNOT

Tähän asti on opeteltu suoraviivaisten JavaScript-ohjelmien tekemistä. Voit tehdä ohjelmiisi rakenteita, jotka ohjaavat niiden suoritusta käyttämällä kontrollikomentoja.

Ehtolauseen avulla voit määrätä ohjelman tekemään tiettyjä toimenpiteitä tiettyjen ehtojen täyttyessä. Ehdot tehdään komennolla **`if`**.

Esimerkki:

```
var a=12;
var b=8;
document.write("A on ");
if (a > b) {
    document.write("suurempi");
} else {
    document.write("pienempi tai yhtäsuuri");
}
document.write(" kuin B.");
```

Jos muuttuja a sisältää suuremman arvoin kuin muuttuja b (kuten esimerkissä), ohjelma tulostaa näytölle "A on suurempi kuin B.". Jos muutat ohjelman alussa olevia lukuja siten että b onkin suurempi kuin a, ohjelman tulostama teksti muuttuu vastaavasti.

Ehtolauseen jälkeen sijoitetaan sulkuihin ehto jota testataan. Esimerkissä se on **`a > b`**, eli *jos a on suurempi kuin b.* Ehtolauseita voidaan koota seuraavilla operaattoreilla:

`a > b`	a on suurempi kuin b
`a < b`	a on pienempi kuin b
`a >= b`	a on suurempi tai yhtä suuri kuin b
`a <= b`	a on pienempi tai yhtä suuri kuin b
`a != b`	a on erisuuri kuin b
`a == b`	a on yhtä suuri kuin b

Ehtolauseilla voidaan testata myös merkkijonoja - niiden tapauksessa ehto *suurempi kuin* tarkoittaa samaa kuin *aakkosissa myöhempänä kuin.* Esimerkiksi "Birgit" on suurempi kuin "Aila".

Ehtolauseen jälkeen merkitään komennot, jotka ajetaan ehtolauseen ollessa tosi. Komentoja voi olla monta - siksi ne kerätään yhteen blokkiin kaarisulkujen (**{** ja **}**) sisään. Tämän blokin jälkeen voidaan sijoittaa vielä ehto **else**, joka tarkoittaa *muussa tapauksessa*. Sen jälkeen tulevan blokin komennot ajetaan, jos ehto ei ollut tosi. **If**-lause näyttää siis tältä:

```
if (ehtolause) {
    nämä komennot ajetaan ehdon ollessa tosi
} else {
    nämä komennot ajetaan jos ehto oli epätosi
}
```

Ehtolauseita voidaan yhdistää monimutkaisemmiksi kokonaisuuksiksi loogisilla operaattoreilla ja sulkumerkeillä. Loogiset operaattorit ovat:

!	Ei
&&	Ja
||	Tai (kaksi pystyviivamerkkiä)

Voit siis tehdä vaikkapa seuraavanlaisen ehdon (*jos kuukausi on joulukuu ja päivä on suurempi kuin 20, tulosta teksti "Joulupukki tulee pian!"*):

```
if ((kuukausi == "joulukuu) && (paiva > 20)) {
    document.write("Joulupukki tulee pian!");
}
```

Ehtojen tulokset antavat tuloksena Boolen arvon. Jos sinulla on muuttuja, joka sisältää Boolen arvoja (tosi tai epätosi), voit testata sitä suoraan **if**-lauseessa:

```
var loma=true;
if (loma) {
 document.write("Nyt ollaan lomalla!");
```

While-silmukalla voit määritellä tietyt komennot toistettaviksi niin kauan, kunnes ehto täyttyy. **While**-komennon jälkeen sijoitetaan ehto, joka testataan. Jos ehto ei täyty, **while** ajaa ehdon jälkeen tulevan blokin ja testaa sitten uudelleen. Tätä toistetaan, kunnes ehto on tosi.

Esimerkki:

```
var kuukausi=1;
while (kuukausi < 13) {
 document.write("Kuukausi "+kuukausi+"<br>");
 kuukausi++;
}
```

Ohjelma alkaa sijoittamalla muuttujaan kuukausi arvon 1. Sitten se toistaa kaarisulkujen sisällä olevaa silmukkaa niin kauan kuin muuttuja kuukausi sisältää arvon 13. Jokaiselle kierroksella ohjelma tulostaa selaimeen tekstin "Kuukausi", muuttujan kuukausi senhetkisen arvon ja rivinvaihdon. Lisäksi se kasvattaa muuttujan kuukausi arvoa yhdellä (muuten ohjelma jäisi ikuiseen silmukkaan jossa se testaisi jatkuvasti että onko 1 pienempi kuin 13).

Ohjelma tulostaa selaimeen seuraavannäköisen HTML-koodin (jonka selain sitten muotoilee käyttäjälle näkyviin):

```
Kuukausi 1
Kuukausi 2
Kuukausi 3
Kuukausi 4
Kuukausi 5
Kuukausi 6
Kuukausi 7
Kuukausi 8
Kuukausi 9
Kuukausi 10
Kuukausi 11
Kuukausi 12
```

Silmukoita voi tehdä myös **for**-komennolla. Tähän komentoon määritellään alkutilanne, loppuehto ja joka kierroksella suoritettava komento. Äskeinen esimerkki voidaan toteuttaa **for**-komennon avulla seuraavasti:

```
for (var kuukausi=1; kuukausi < 13; kuukausi++) {
 document.write("Kuukausi "+kuukausi+"<br>");
}
```

For-komennossa ensimmäisenä määritellään muuttujan alkuarvo. Puolipisteen jälkeen tulee ehto, joka tarkistetaan ennen jokaista kierrosta ja viimeisenä jokaisen kierroksen lopuksi ajettava komento.

FUNKTIOT

Usein käytetyistä ohjelman osista voi tehdä funktion. Funktiot ovat ikään kuin uusia komentoja: voit kutsua niitä myöhempänä ohjelmassa käyttämällä funktion nimeä.

```
function ylistys ( ) {
 document.write("Olet maailman paras ");
 document.write("Javascript-ohjelmoija!");
}
ylistys();
ylistys();
```

Yllä oleva esimerkki määrittelee funktion **ylistys**. Komennon **function** jälkeen annetaan suluissa mahdolliset funktion käyttämät parametrit (näistä lisää alempana) ja sen jälkeen kaarisulkujen sisällä varsinaiset komennot. Esimerkin funktio tulostaa kaksi riviä tekstiä.

Myöhempänä ohjelmassa funktiota kutsutaan yksinkertaisesti sen nimellä. Funktion nimen perässä luetellaan sen käyttämät parametrit (näistä lisää myöhemmin). Esimerkin funktio ei käytä parametrejä, joten sen perään sijoitetaan tyhjät sulut. Esimerkin ohjelmanpätkä tulostaa kaksi kertaa ylistystekstin "**Olet maailman paras Javascript-ohjelmoija!** ".

Funktiolle voidaan antaa parametrejä, joiden avulla funktio voi muuttaa toimintaansa. Funktion käyttämät parametrit annetaan sulkujen sisällä:

```
function ylistys(nimi) {
 document.write("<p>"+nimi+" on maailman ");
 document.write("paras ohjelmoija!");
}
```

Kun funktiota kutsutaan, sille annettu parametri sijoitetaan muuttujaan **nimi**. Tämän jälkeen sitä voidaan käyttää funktion sisällä kuten muitakin muuttujia:

```
ylistys("Sami");
```

Esimerkki tulostaa selaimelle tekstin:

```
<p>Sami on maailman paras ohjelmoija!
```

Funktion kutsussa voidaan käyttää myös muuttujia:

```
var kuka="Riitta";
ylistys(kuka);
```

Funktio voi saada useampia parametrejä ja palauttaa arvoja. Voit tehdä funktion, joka laskee nettopalkkion saatuaan parametreinä bruttopalkkion ja veroprosentin:

```
function laskenetto(brutto, veropr) {
 var netto=brutto - brutto*(veropr/100);
 return (netto);
}
```

Funktio palauttaa laskennan tuloksen komennolla **return**.

```
var palkio=4000;
var vero=36;
document.write("Nettopalkkiosi on ");
var netto=laskenetto(palkio,vero);
document.write(netto+" mk");
```

Palautearvoa käytetään sijoittamalla funktion tulos suoraan muuttujaan tai tulostamalla se. Yllä olevassa esimerkissä funktion palauttama tulos sijoitetaan muuttujaan **netto** ja tulostetaan sitten selaimeen **document.write**-komennolla:

```
Nettopalkiosi on 2560 mk
```

Jos muutat veroprosenttia tai bruttopalkkiota, voit katsella miten ne vaikuttavat ohjelman tulosteisiin. Funktiota voidaan käyttää myös suoraan, ilman muuttujia:

```
document.write(laskenetto(10000,12));
```

OLIOT

JavaScript on olioperustainen ohjelmointikieli. Voit rakentaa JavaScript-ohjelmien sisällä olioita tai käyttää kielessä valmiina olevia olioita hyväksesi. Oliot ovat eräänlaisia muuttujien ja funktioiden yhdistelmiä - tietotyyppejä, jotka voivat sisältää sekä muuttujia että funktioita.

JavaScript-ohjelma voi sisältää esimerkiksi olion **auto**, joka sisältää tietoja autosta. Olion sisältämiä tietoja voivat olla esimerkiksi väri ja hinta. Olion sisältämiä muuttujia kutsutaan *attribuuteiksi*.

Voit asettaa ja lukea olion attribuutteja antamalla ensin olion nimen, pisteen ja sen jälkeen attribuutin nimen:

```
auto.vari="Punainen";
document.write(auto.hinta);
```

Olio voi sisältää myös funktioita, joita kutsutaan *metodeiksi*. Metodit toimivat kuten tavallisetkin funktiot: niille annetaan parametrejä ja ne voivat palauttaa arvoja. Auto-olion metodit voivat esimerkiksi kertoa, miten paljon auton tankissa on bensiiniä:

```
document.write(auto.bensatilanne());
```

Olio luodaan tekemällä sille luontifunktio, *konstruktori*. Tämä funktio rakentaa olion ja määrittelee sen attribuutit ja metodit. Se on malli, jonka perusteella uudet oliot rakennetaan. Konstruktori saa parametreinä alkuarvot jotka sijoitetaan uuteen olioon. Alla näet esimerkin auto-olion konstruktorista:

```
function auto(vari, hinta, malli) {
 this.vari=vari;
 this.hinta=hinta;
 this.malli=malli;
}
```

Konstruktorin sisällä käytetään viittausta **this** osoittamaan luotavana olevaa oliota. Uusi auto luodaan konstruktorin ja **new**-komennon avulla:

```
ford=new auto("Punainen",79000,"Fiesta");
mersu=new auto("Musta",250000,"C-luokka");
```

Voit käyttää myöhemmin ohjelmassa olion attribuutteja ja asettaa niitä:

```
document.write(ford.vari);
// tulostaa "punainen"
mersu.hinta=300000;
// nostetaan mersun hintaa
```

Voit tehdä olion sisään myös funktioita, eli metodeita. Ensin tehdään erillinen funktio, joka sitten liitetään olion konstruktoriin:

```
function kaverialennus(hinta) {
 // kaverille auto myydään 80%
 // alkuperäisestä hinnasta
 return hinta*0.80;
}

function auto(vari, hinta, malli) { this.hinta=hinta;
 this.vari=vari;
 this.malli=malli;
 this.kaverinhinta=kaverialennus(this.hinta);
}
```

Funktion määrittelyn sisällä käytetään nimeä **this** viittaamaan käsittelyn alla olevaan olioon. Funktio **kaverialennus** on normaali funktio, joka laskee annetusta parametristä 80% ja palauttaa sen. Tämä funktio liitetään konstruktoriin siten, että kutsuttaessa olion metodia **kaverinhinta** olio laskee sen funktiolla **kaverialennus** antaen parametriksi oliossa määritellyn hinnan.

Olion metodia käytetään kuten tavallisiakin funktioita:

```
ford=new auto("Punainen",79000,"Fiesta");
document.write("Parhaalle kaverille ");
document.write(ford.malli+" myydään ");
document.write("erikoishintaan ");
document.write(ford.kaverinhinta);
document.write(" mk");
```

Esimerkkiohjelma tulostaa ruudulle tekstin:

```
Parhaalle kaverille Fiesta myydään erikoishintaan
63200 mk
```

SISÄÄNRAKENNETUT OLIOT

JavaScript sisältää joukon sisäänrakennettuja olioita, joiden avulla voit rakentaa itsellesi erilaisia tietorakenteita.

Jos haluat tehdä itsellesi taulukon, voit tehdä itsellesi **Array**-olion. Taulukon konstruktorille voidaan antaa parametriksi joko taulukon koko tai taulukon sisältöön viittaavia arvoja:

```
taulukko=new Array(10);
vuodenajat=new Array("Kevät", "Kesä", "Syksy",
"Talvi");
```

Voit sijoittaa ja lukea taulukon muuttujia sijoittamalla nimen perään taulukon indeksin:

```
taulukko[1]=15;
taulukko[3]=taulukko[2]+8;
vuodenajat["Syksy"]="Inhottava";
```

Voit luoda päivämäärä-olioita konstruktorilla **Date**. Konstruktorille voidaan antaa parametriksi haluttu päivämäärä joko tekstimuodossa tai numeroina, tai tyhjä arvo jolloin olion sisällöksi tulee kuluva hetki.

Esimerkkejä:

```
nyt=new Date();
joulu=new Date("December 24, 1997 18:00:00");
juhannus=new Date(97,6,24,21,0,0);
```

Tekstimuodossa päivämäärän sisällössä annetaan kuukauden nimi englanniksi, päivän numero, pilkku, vuoden numero ja kellonaika jossa tunnit, minuutit ja sekunnit erotellaan kaksoispisteellä. Kellonaika voidaan jättää pois, jolloin siihen sijoitetaan keskiyö.

Numeroina ajat annetaan numeroina peräkkäin samassa järjestyksessä.

Päivämääräoliolla on useita metodeja, joilla voit purkaa päivämäärän sisältämiä aikoja. Näitä ovat **getTime**, **getDay**, **getHours**, **getMinutes**, **getMonth**, **getSeconds**, **getTime**, **getTimeZoneOffset** ja **getYear**. Voit lukea esimerkiksi ajasta vuoden seuraavasti:

```
jouluvuosi=joulu.getYear;
```

Metodit **parse** ja **getTime** palauttavat kuluneiden millisekuntien määrän uudesta vuodesta 1970.

Voit muuttaa aikaa metodeilla **setDate**, **setHours**, **setMinutes**, **setMonth**, **setSeconds**, **setTime** ja **setYear**. Voit muuttaa esimerkiksi ajan **joulu** tunteja seuraavasti:

```
joulu.setHours(19);
```

Metodille **setTime** annetaan parametri millisekunteina uudesta vuodesta 1970.

Voit muuttaa ajan antamaan vastaavan ajan GMT-aikavyöhykkeen aikana metodilla **toGMTString**. Metodi **toLocaleString** palauttaa ajan paikallisessa esitysmuodossa.

Metodilla **UTC** voit kysyä aikaleimaa (sekunteja uudesta vuodesta 1970) mille tahansa ajalle antamalla parametrit numeroina. Parametrit annetaan samalla tavalla kuin **Date**-olion konstruktorissa. Tätä metodia voi käyttää suoraan **Date**-olion kautta:

```
aikaleima=Date.UTC(97,11,4,17,40,53);
```

Olio **Math** sisältää matemaattisia toimintoja. Voit käyttää niitä suoraan olion kautta:

```
x=Math.sin(1.23);
```

Matemaattisia metodeja ovat:

abs (itseisarvo)
sin, **cos**, **tan** (sini, kosini ja tangentti - anna kulmat radiaaneina)
acos, **asin**, **atan** (arkuskosini, arkussini ja arkustangentti)
exp, **log** (eksponenttifunktio ja luonnollinen logaritmi)
ceil (seuraava kokonaisluku)
floor (edellinen kokonaisluku)
min, **max** (suurempi tai pienempi kahdesta parametristä)
pow (potenssiin korottaminen)
round (kokonaislukuun pyöristys)
sqrt (neliöjuuri)

Merkkijonojen käsittelyyn on käytössä **String**-olio. Voit luoda uuden merkkijono-olion antamalla konstruktorin parametriksi merkkijonon:

```
kaupunki=new String("Tampere");
```

Saat selville merkkijonon pituuden attribuutista **length**:

```
pituus=kaupunki.length;
```

Muuttuja **pituus** saa esimerkissä arvon 7.

Merkkijonoja voidaan käsitellä seuraavilla metodeilla:

charAt	Palauttaa parametrinä annetun merkin paikan jonossa.
indexOf	Palauttaa ensimmäisen parametrinä annetun merkkijonon paikan merkkijonon sisältä.
lastIndexOf	Kuten yllä, mutta palauttaa viimeisen löydetyn paikan.
split	Paloittelee merkkijonon osiksi parametrinä annettujen kirjaimien kohdalta. Palauttaa arvona taulukon merkkijonon paloista.

substring	Saa arvonaan kaksi numeroa ja palauttaa merkkijonosta näiden väliin jäävän osan. Esimerkiksi **kaupunki.substring(0,3)** palauttaa "Tam".
toLowerCase	Muuttaa merkkijonon kaikki kirjaimet pieniksi.
toUpperCase	Muuttaa merkkijonon kaikki kirjaimet suuriksi.

Merkkijono voidaan tulostaa HTML-komennoilla ympäröitynä käyttäen attribuutteja **anchor**, **big**, **blink**, **bold**, **fixed**, **italics**, **small**, **strike**, **sub**, **sup** tai **link**.

Muita sisäänrakennettuja olioita ovat **Number** (numeroiden käsittely), **Function** (tekee funktion) ja **Boolean** (tekee Boolen arvon). Alla näet esimerkkejä näiden käytöstä:

```
totuusarvo=new Boolean(false);
tulostusfunktio=new Function("document.write ('Heippa,
maailma')");
suurinumero=Number.MAX_VALUE;
pieninnumero=Number.MIN_VALUE;
aareton=Number.POSITIVE_INFINITY;

tulostusfunktio();
```

OLIOT JA WWW-SIVUT

JavaScript-ohjelmat voivat vaikuttaa WWW-sivuihin ja selaimiin käsittelemällä valmiiksi määriteltyjä olioita.

DOCUMENT-OLIO

Yksi valmiiksi määritellyistä olioista on **document**. Se sisältää useita attribuutteja ja metodeita, joita voit käyttää hyväksesi JavaScript-ohjelmissa. Olet jo tutustunut **document**-olion metodiin **write**, jolla voi tulostaa selaimelle HTML-koodia:

```
document.write("<p>Hei, maailma!");
```

Document-olio sisältää monia attribuutteja ja metodeja, jotka liittyvät näytöllä olevaan sivuun. Voit lukea ja asettaa sivujen värejä käyttäen hyväksesi väriattribuutteja:

document.bgColor	Taustaväri
document.fgColor	Tekstin väri
document.linkColor	Linkin väri
document.vlinkColor	Käydyn linkin väri
document.alinkColor	Aktivoidun linkin väri

Attribuutit voivat saada arvonsa samalla tavalla kuin **body**- tai **font**-komennoissa annettavat attribuutit:

```
document.bgColor="blue";
document.fgColor="#FFFFFF";
// valkoista tekstiä sinisellä taustalla
```

Huomaa, että jos määrittelet sivun taustavärin ennen **body**-komentoa, taustavärin vaihtaminen ei toimi sillä **body**-komento muuttaa myös sivun taustavärin!

Document-olio sisältää myös linkkeihin liittyviä attribuutteja:

document.location	sivun URL
document.referrer	edellisen sivun URL

Attribuutti **location** sisältää näytöllä olevan sivun URL-osoitteen. Jos sivulle on saavuttu linkin kautta, attribuutti **referrer** sisältää edellisen sivun osoitteen. Voit lukea attribuutteja:

```
document.write("Tulit tänne sivulta ");
document.write(document.referrer);
```

Attribuutteihin voidaan myös sijoittaa arvoja. Tällä tavalla saat selaimen hyppäämään tietylle sivulle:

```
document.location="http://www.cnn.com/";
```

Sivulla olevat kuvat ja lomakkeiden osat kuuluvat osana **document**-olioon. Kun annat kuville tai lomakkeiden osille nimen **name**-attribuutilla, voit käsitellä niitä **document**-olion attribuutteina:

```
<img src="pallo.gif" name="pallokuva">
```

Kun on lisännyt HTML-sivulle kuvan yllä olevalla komennolla, voit vaihtaa myöhemmin kuvaa määrittelemällä uudelleen sen **src**-attribuutin:

```
document.pallokuva.src="pallo2.gif";
```

Kuvaoliolla ovat samat attribuutit, jotka voi asettaa HTML-komennoissakin.

Myös sivulla olevat lomakkeet voidaan nimetä:

```
<form method=POST action="mailto:minä@jossain"
name="omalomake">
<p>Anna sukunimesi: <input type=text name="sukunimi">
</form>
```

Voit lukea tai asettaa lomakkeen tekstilaatikossa olevan tekstin käsittelemällä **document**-olion attribuuttia **omalomake**:

```
document.omalomake.sukunimi.value="Virtanen";
```

Tässä esimerkissä muutettiin lomakkeessa **omalomake** olevaa tekstilaatikkoa **sukunimi** asettamalla sen sisällöksi (**value**) "Virtanen".

Muita **document**-olion attribuutteja ovat:

anchors	Taulukko, joka sisältää sivulla olevat linkit ja kirjanmerkit.
applets	Taulukko, joka sisältää sivulla olevat Java-appletit.
cookie	Sisältää sivun **cookie**-muuttujan arvon.

domain	Sisältää Internet-osoitteen domain-osan sille palvelimelle, josta sivu on haettu.
embeds	Sisältää taulukon sivulla olevista **plug-in**-objekteista.
forms	Sisältää taulukon sivulla olevista lomakkeista.
images	Sisältää taulukon sivulla olevista kuvista.
lastModified	Sisältää päivämäärän, jolloin dokumenttia on viimeksi muutettu.
links	Sisältää taulukon sivulla olevista linkeistä.
title	Sisältää sivun nimen.
URL	Sisältää sivun URL-osoitteen.

Document-olio sisältää lisäksi seuraavat metodit:

open	Avaa dokumentin kirjoitusta varten.
close	Sulkee dokumentin ja tulostaa dokumenttiin kaikki siihen kirjoitetut tekstit.
eval	Ajaa parametrinä annetun JavaScript-koodin. Tällä tavalla voit kerätä komentoja merkkijonomuuttujaan ja ajaa niitä myöhemmin.
toString	Muuttaa parametrinä annetun olion merkkijonomuuttujaksi.
valueOf	Palauttaa parametrinä annetun olion arvon.
write	Tulostaa tekstiä sivulle. Parametrinä annetaan tulostettava teksti.
writeln	Kuten **write**, mutta tulostaa tekstin jälkeen rivinvaihdon.

WINDOW-OLIO

Window-olio sisältää selaimen ikkunaan liittyviä ominaisuuksia:

window.location	Ruudulla näkyvän ikkunan URL
window.status	Alareunan status-palkin teksti

Voit asettaa selaimen ikkunan alareunassa näkyvän tekstin:

```
window.status="Tervetuloa!";
```

Huomaa kuitenkin, että selain saattaa kirjoittaa status-riville omia tekstejään (ja poistaa sinne sijoittamasi tekstin).

Window-olio sisältää myös joukon metodeja, joita voit käyttää hyväksesi ohjelmissasi.

window.open	Avaa uuden selainikkunan.
window.close	Sulkee selainikkunan
window.alert	Tuo näytölle ilmoitusikkunan
window.confirm	Tuo näytölle varmistusikkunan
window.prompt	Pyytää käyttäjältä syötettä
window.focus	Tekee ikkunasta aktiivisen
window.blur	Poistaa ikkunan aktivoinnin
window.scroll	Vierittää ikkunaa
window.setTimeout	Tekee operaation tietyn ajan kuluttua

Voit avata uuden ikkunan käyttämällä **window.open** -metodia. Metodille annetaan parametriksi tiedostonnimi tai URL-osoite, josta ikkunan sisältö haetaan ja mahdollinen ikkunalle annettava nimi. Ikkunan nimeä voidaan käyttää esimerkiksi linkkien kohteina (linkkikomennon **target**-attribuutti).

Alla oleva esimerkki avaa uuden ikkunan, jonka sisältö tulee osoitteesta **http://www.yahoo.com**. Ikkunan nimeksi annetaan "**yahooikkuna**".

```
uusi=window.open("http://www.yahoo.com/",
"yahooikkuna");
```

Komento luo uuden objektin nimeltään **uusi**. Voit käyttää tätä objektia samalla tavalla kuin muitakin **window**-objekteja - lukea ja asettaa sen attribuutteja tai kutsua sen metodeja. Voit esimerkiksi sulkea juuri avaamasi ikkunan kutsumalla sen **close**-metodia:

```
uusi.close();
```

Jos haluat sulkea aktiivisen ikkunan, voit käyttää jompaa kumpaa seuraavista komennoista:

```
window.close();
self.close();
```

Voit antaa ikkunalle avauskomennossa joukon attribuutteja, jotka kertovat lisää avattavan ikkunan ominaisuuksista. Attribuutteja ovat:

toolbar	Onko ikkunassa näkyvissä painikerivi (**Back**- ja **Forward**-painikkeet ym.)?
location	Onko ikkunassa näkyvissä osoitepalkki?
directories	Onko ikkunassa näkyvissä **Directories**-rivi (**What's New**- ja **What's Cool** -painikkeet ym.)?
status	Onko ikkunassa näkyvissä alareunan **status**-rivi?
menubar	Onko ikkunan yläreunassa näkyvissä valikkopalkki?
scrollbars	Onko ikkunassa näkyvissä vierityspalkit?
resizable	Voiko ikkunan kokoa muuttaa?

width	Ikkunan leveys pikseleinä.
height	Ikkunan korkeus pikseleinä.

Kaikki muut attribuutit paitsi **width** ja **height** saavat joko arvon **yes** (tai **1**) tai **no** (tai **0**). Arvot kytkevät ominaisuuden päälle tai pois päältä. Attribuutit lisätään omaksi parametrikseen uuden ikkunan aloitusmetodiin. Alla näkyvällä tavalla voit avata uuden ikkunan, joka ei sisällä muita kohtia kuin pelkän sisällön. Ikkunan kooksi määritellään 300 x 200 pikseliä.

```
uusi=window.open("http://www.yahoo.com/",
"yahooikkuna", "toolbar=no,location=no,directories=no,
status=no,menubar=no,scrollbars=no,resizable=no,
width=300,height=200");
```

Huomaa, kuinka ikkunan attribuutit ovat yhden parametrin sisällä. Älä sijoita parametrin sisään välilyöntejä.

Voit tuoda käyttäjälle valintaikkunan **window**-objektin metodilla **confirm**. Metodi tuo käyttäjälle näkyviin ikkunan, jossa on valintapainikkeet **OK** ja **Cancel**. Metodi palauttaa arvon **true** tai **false** riippuen siitä, mitä painiketta käyttäjä napsautti. Metodille voi antaa parametriksi ikkunaan tulostettavan tekstin:

```
vastaus=window.confirm("Oletko varma?");
if (vastaus == false) {
 document.write("Ei sitten");
} else {
 document.write("Hienoa!");
}
```

Voit pyytää käyttäjältä tekstiä metodilla **prompt**. Metodi palauttaa arvona syötetyn tekstin:

```
nimi=window.prompt("Mikä on nimesi?");
document.write("Hei, "+nimi);
```

Metodilla **setTimeout** voit määritellä funktioita ajettavaksi tietyn ajan kuluttua. Metodille annetaan kaksi parametriä: funktion nimi ja aika, jonka

kuluttua funktio ajetaan. Aika annetaan millisekunteina (sekunnin tuhannesosina). Esimerkiksi:

```
function halytys() {
 window.alert("Olet lukenut tätä sivua jo 10
sekuntia!");
}
timerID=setTimeout("halytys()", 10000);
```

Funktio palauttaa tunnisteen (esimerkissä **timerID**), jonka avulla laskuri voidaan peruuttaa käyttäen **clearTimeout**-metodia:

```
clearTimeout(timerID);
```

Window-olio on hierarkiassa ylempänä kuin **document**-, **location**- ja **history**-oliot. Jos haluat käsitellä toisessa ikkunassa olevaa **document**-oliota, voit viitata siihen ikkunan olion kautta:

```
uusi.document.write("tekstiä");
```

LOCATION-OLIO

Location-olio sisältää tietoa ladatusta URL-osoitteesta. Se on osa **window**-oliota, joten voit käyttää **location**-oliota suoraan tai **window**-olion kautta (jos viittaat toisen ikkunan osoitteisiin):

```
location.href;
window.location.href;
uusi.location.href;
```

Kaksi ylintä viittaavat aktiivisen ikkunan osoitteeseen ja alin **uusi**-ikkunan osoitteeseen.

Olio sisältää seuraavat attribuutit:

protocol	Osoitteen alkuosa eli protokolla (yleensä **http:**).
host	Tietokoneen nimi.

port	Osoitteessa mahdollisesti määriteltynä oleva porttinumero.
pathname	Osoitteessa oleva polku ja tiedostonnimi.
hash	Osoitteessa mahdollisesti oleva kirjanmerkki (**#**-merkin jälkeen tuleva osa).
search	Osoitteessa mahdollisesti oleva hakusana/parametri (**?**-merkin jälkeen tuleva osa).
href	URL-osoite kokonaisuudessaan.
hostname	Osoitteessa oleva tietokoneen nimi porttinumeroineen (**nimi:portti**).

Location-olio sisältää seuraavat metodit:

reload	Pakottaa sivun uudelleenlatauksen (kuten **Reload**-painikkeen napsautus).
replace	Lataa uuden sivun vanhan tilalle siten, että vanhalle sivulle ei pääse **Back**-painikkeella.

HISTORY-OLIO

History-olio sisältää tietoa edellisistä sivuista. Olio on **window**-olion sisällä, joten voit käyttää sitä kuten **location**-oliota.

Voit viitata **history**-olioon taulukkona, jolloin pääset käsiksi osoitteisiin joissa käyttäjä on käynyt. Esimerkiksi **history[1]** sisältää viimeistä edellisen käydyn sivun osoitteen.

Oliolla on seuraavat attribuutit:

current	Tämänhetkinen osoite.
length	Historiataulukon pituus.
next	Seuraava osoite.

previous Edellinen osoite.

Voit käyttää myös seuraavia metodeja:

go Siirtyy tiettyyn käytyyn osoitteeseen. Parametrinä annetaan siirtymien lukumäärä, esimerkiksi **history.go(-3)** siirtyy kolme pykälää taaksepäin.

back Siirtyy edelliseen osoitteeseen (vastaa **Back**-painikkeen napsautusta).

forward Siirtyy seuraavaan osoitteeseen (vastaa **Forward**-painikkeen napsautusta).

NAVIGATOR-OLIO

Navigator-olio sisältää tietoa selaimesta. Se sisältää seuraavat attribuutit:

appCodeName Selaimen koodinimi. Netscapessa tämä nimi palauttaa tekstin "Mozilla".

appName Palauttaa selaimen nimen, esimerkiksi "Netscape".

appVersion Palauttaa selaimen versionumeron.

mimeTypes Palauttaa taulukon selaimen tukemista mediatyypeistä.

plugins Palauttaa taulukon selaimeen asennetuista **plugineistä**.

userAgent Palauttaa tekstin, joka kertoo tietoja selaimesta ja käyttöjärjestelmästä.

Navigator-oliolla on seuraavat metodit:

javaEnabled Palauttaa arvon **true**, jos selain tukee Javakieltä.

taintEnabled	Palauttaa arvon **true**, jos selaimessa on asetettu päälle turvallisuustoiminto, joka estää salaisen tiedon siirtämisen palvelimelle.

TAPAHTUMAT

JavaScriptin avulla voi tarkkailla käyttäjän toimenpiteitä, kuten hiiren osoittimen siirtämistä, kirjoittamista ja navigointia. Toimenpiteet aiheuttavat tapahtumia (**event**), joita JavaScript-ohjelmat voidaan määrätä tarkkailemaan.

Tarkkailu määritellään lisäämällä HTML-komentoihin tapahtumien käsittelijöitä (**event handler**). Nämä komennot eivät siis tule **script**-komentojen väliin.

Voit tehdä esimerkiksi linkkiin tapahtumankäsittelijän, joka aktivoituu kun käyttäjä siirtää hiiren osoittimen linkin päälle. Tämä tapahtuu lisäämällä linkkikomentoon JavaScript-attribuutti **onMouseOver**.

Voit lisätä tapahtumankäsittelijään vaikkapa komennon, joka asettaa **window**-olion **status**-attribuuttiin (selaimen ikkunan alareunan tekstiksi) jonkin tekstin:

```
<a href="http://www.cnn.com/"
onMouseOver="window.status='CNN';
return true;">
Tästä CNN:n kotisivulle</a>
```

Huomaa, että attribuutin **onMouseOver** sisältönä olevat JavaScript-komennot sijoitetaan lainausmerkkeihin:

```
onMouseOver="komentoja"
```

Jos sisällä olevissa komennoissa käytetään lainausmerkkejä, niiden sijasta käytetään heittomerkkejä (kuten muistat, JavaScript-komennoissa voi käyttää kumpia tahansa). Tällä tavalla ulommat lainausmerkit liittyvät **onMouseOver**-komentoon ja sisemmät heittomerkit liittyvät sen sisällä oleviin JavaScript-komentoihin (esimerkissä **window.status='teksti'**).

Lainausmerkkien sisällä voi olla useampia komentoja puolipisteellä eroteltuna (aivan kuten JavaScript-ohjelmissa yleensäkin). Esimerkissä ikkunan alareunan tekstin asettaminen vaatii, että **`window.status`**-attribuutin asettamisen jälkeen annetaan komento **`return true`**.

Tapahtumienkäsittelijöitä on monia: jotkut liittyvät hiiren liikkeisiin, jotkut lomakkeisiin ja jotkut ovat yleiskäyttöisempiä.

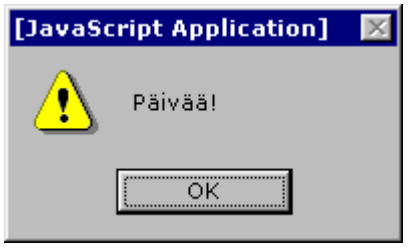
`onLoad`	Aktivoituu sivun latautuessa (sijoita **`body`**-komentoon).
`onUnload`	Aktivoituu poistuttaessa sivulta (sijoita **`body`**-komentoon).
`onMouseOver`	Kun hiiren osoitin tulee linkin, kirjanmerkin tai objektin ylle.
`onMouseOut`	Kun hiiren osoitin poistuu linkin, kirjanmerkin tai objektin päältä.
`onClick`	Kun jotain yllämainituista napsautetaan.
`onFocus`	Kun tekstilaatikko tai muu lomakekenttä aktivoidaan (esimerkiksi napsautetaan hiirellä tekstikentässä niin että siihen voidaan kirjoittaa).
`onBlur`	Kun aktivointi katoaa yllämainituista.
`onChange`	Kuten yllä, mutta sisältö on muuttunut.
`onSelect`	Kun lomakkeessa maalataan tekstiä.
`onSubmit`	Kun lomake lähetetään **`submit`**-painikkeella.
`onAbort`	Kun käyttäjä keskeyttää kuvien latauksen painamalla **`Stop`**-painiketta tai valitsemalla linkin.
`onError`	Kun sivun lataamisessa tapahtui virhe.
`onReset`	Kun käyttäjä painaa lomakkeen **`reset`**-painiketta.

JavaScript sisältää komennon **alert**, jolla voit avata käyttäjälle pienen ikkunan, joka sisältää haluamasi viestin. Voit käyttää tätä hyväksesi esimerkiksi testatessasi tapahtumien käsittelijöitä.

Alert-komennolle annetaan parametriksi haluttu viesti:

```
alert("Päivää!");
```

Selain näyttää ilmoituksen erillisessä ikkunassa:



Voit kytkeä ilmoituksen esimerkiksi sivulta poistumiseen:

```
<body onUnload="alert('Hyvää matkaa!')">
```

Jos haluat ajaa tapahtumankäsittelijän yhteydessä useampia komentoja, sijoita komennot erilliseen funktioon ja laita tämä funktio tiedoston alkuun **head**-komentojen väliin. Tällä tavalla sinun ei tarvitse sijoittaa itse HTML-komentojen yhteyteen pitkiä komentosarjoja.

Tehdään esimerkki, jossa sivun taustaväri vaihtuu punaiseksi, kun käyttäjä vie hiiren osoittimen linkin päälle ja palautuu siniseksi, kun osoitin siirretään muualle.

```
<html>
<head><title>Javascriptiä</title>
<script language="JavaScript">
<!--
function punaiseksi() {
 document.bgColor="red";
}

function siniseksi() {
 document.bgColor="blue";
}
```

```
// -->
</script>
</head>
<body onLoad="siniseksi()" text="White" link="Yellow"
vlink="Yellow">

<p>Linkkejä:
<ul>
<li><a href="http://www.cnn.com/"
onMouseOver="punaiseksi()"
onMouseout="siniseksi()">CNN</a>
</ul>
</body> </html>
```

Alussa määritellään kaksi funktiota: **punaiseksi** ja **siniseksi**, jotka muuttavat sivun taustan värin. Kun sivu ladataan, sen taustaväriksi asetetaan sininen, tekstin väriksi valkoinen ja linkkien väriksi keltainen (nämä erottuvat sekä punaiselta että siniseltä taustalta).

Itse linkkikomentoon liitetään tapahtumankäsittely sille, kun hiiren osoitin siirtyy kuvan päälle tai poistuu kuvan päältä, ja muutetaan sivun taustaväriä aikaisemmin määriteltyjen funktioiden avulla.

Samantyyppisellä funktiolla voit tehdä sivun, jonka kuvat vaihtuvat toisiksi kun hiiren osoitin viedään kuvan päälle. Kuvan vaihtaminenhan tapahtui antamalle kuvalle nimi, jolloin voit käsitellä sitä **document**-olion attribuuttina. Muuttamalla kuvan **src**-attribuuttia voit vaihtaa sitä, mikä kuva on näkyvillä ruudulla.

JAVASCRIPT JA TASOT

JavaScriptin ja tasojen avulla saat tehtyä sivuillesi näppärästi monipuolisia toimintoja. Jokainen taso näkyy JavaScript-ohjelmalle omana olionaan.

Kun olet antanut tasolle nimen, voit käyttää sitä olion **document.*tasonnimi*** kautta. Voit muuttaa tason kokoa, sen näkyvyyttä tai mitä tahansa sen ominaisuuksista.

Alla on esimerkki, jossa määritellään näkymätön taso nimeltään **ohje**. Sivulla olevaan linkkiin kytketään tapahtumankäsittelijä, joka tuo tason näkyviin silloin kun hiiren osoitin siirtyy linkin päälle.

```
<html>
<head>
<title>Javascript ja tasot</title>
</head>
<body>
<layer id=ohje top="10%" left="30%" width=100
height=50 bgcolor="Yellow" visibility=hidden>
<p>Klikkaa tästä niin näet Yahoon kotisivun!
</layer>
<h1>Javascript ja tasot</h1>
<p>Linkki
<a href="http://www.yahoo.com/"
onMouseOver="document.ohje.visibility='show'; return
true;" onMouseOut="document.ohje.visibility='hide';
return true;"> Yahoon</a> sivuille.
</body>
</html>
```

Kun sivu ladataan, taso joka sisältää ohjeen ei ole näkyvissä, sillä **layer**-komennossa on annettu attribuutille **visibility** arvo **hidden**.

Kun hiiren osoitin viedään **Yahoo**-linkin kohdalle, JavaScript-ohjelma asettaa **ohje**-tason attribuuttiin **visibility** arvon **show**, jolloin ko. taso tulee näkyviin. Kun hiiren osoitin viedään pois, **onMouseOut**-tapahtumankäsittelijä vaihtaa tason näkyvyydeksi jälleen **hide**.

Taso-olioilla on käytössä alla näkyvät attribuutit. Monet attribuuteista vastaavat **layer**-komennon attribuutteja ja voivat saada samoja arvoja.

- **document**: Tämän attribuutin kautta voidaan käsitellä tason sisällä olevia kuvia, lomakkeita ja muita olioita. Tätä arvoa ei voi muuttaa.

- **name**: Tason nimi. Tätä arvoa ei voi muuttaa.

- **left**: Tason vasemman yläkulman sijaintipaikka vaakasuunnassa. JavaScript-ohjelmat voivat muuttaa tätä arvoa, jolloin taso liikkuu vastaavasti ruudulla. Vastaava arvo pystysuunnalle on **top**. Jos taso on toisen tason sisällä, etäisyys lasketaan edellisen tason vasemmasta yläkulmasta. Absoluuttinen sijainti sivun vasempaan ylänurkkaan nähden saadaan attribuuteista **pagex** ja **pagey**.

- **zIndex**: Tämä attribuutti sisältää tason sijainnin muihin tasoihin nähden. Sisältöä muuttamalla saat tuotua tason muiden tasojen päälle tai sijoitettua sen niiden alle. Vastaa **layer**-komennon vastaavaa attribuuttia.

- **visibility**: Määrittelee tason näkyvyyden. Arvo **show** kertoo, että taso on näkyvissä, **hide** että taso ei näy.

- **clip.top**, **clip.left**, **clip.right**, **clip.bottom**, **clip.width** ja **clip.height**: Määrittelevät tason näkyvän osan. Tasoa voidaan leikata muuttamalla näiden attribuuttien arvoja. Arvot määrittelevät leikkauksen yläreunan, vasemman reunan, oikean reunan, alareunan, leveyden tai korkeuden.

- **background**: Tason taustakuva.

- **bgColor**: Tason taustaväri.

- **siblingAbove**: Jos taso on toisen tason peittämä, päällä olevaa tasoa voidaan käsitellä tästä attribuutista. Vastaava attribuutti alla olevalle tasolle on **siblingBelow**.

- **above**: Kuten yllä, mutta ottaa huomioon myös tämän tason sisällä määritellyt tasot. Vastaava attribuutti alla olevalle tasolle on **below**.

- **parentLayer**: Jos tämä taso on toisen tason sisällä, edelliseen tasoon päästään käsiksi tästä attribuutista.

- **src**: URL, joka osoittaa tason sisältöön.

Taso-olio sisältää myös useita metodeita, joiden avulla voit käsitellä tasoa. Metodeita käsitellään taso-olion kautta, esimerkiksi **document.ohjetaso.moveBy(20,20)** .

- **moveBy(vaaka,pysty)**: Siirtää tasoa annetun pikselimäärän. Muuttuja **vaaka** sisältää siirtomäärän oikealle päin (anna negatiivinen arvo jos haluat siirtää vasemmalle) ja **pysty** siirtomäärän alaspäin (negatiivinen arvo siirtää ylöspäin).

- **moveTo(x,y)**: Siirtää tason annettuun paikkaan. Vastaa arvojen sijoittamista **top**- ja **left**-attribuutteihin. Jos taso on toisen tason sisällä, koordinaattien origo on edellisen tason vasemmassa yläkulmassa.

- **moveToAbsolute(x,y)**: Siirtää tason annettuun paikkaan. Koordinaatit kertovat paikan suhteessa sivun vasempaan ylänurkkaan.

- **resizeBy(vaaka,pysty)**: Kasvattaa tason kokoa vaaka- ja pystysuunnassa annetun pikselimäärän verran. Tason sisältö ei muutu koonmuutoksen aikana.

- **resizeTo(vaaka,pysty)**: Muuttaa tason annetun kokoiseksi. Tason sisältö ei muutu koonmuutoksen aikana.

- **moveAbove(taso)**: Siirtää tason z-järjestyksessä toisen kehyksen yläpuolelle. Parametri sisältää toisen tason olion. Vastaava metodi alle siirtämiseen on **moveBelow(taso)**.

- **load(osoite,leveys)**: Lataa tason sisään uuden tiedoston ja muuttaa samalla sen leveyttä annettuun pikselimäärään.

Voit luoda uuden tason JavaScriptissä seuraavasti:

```
minuntasoni=document.minuntasoni;
uusitaso=new Layer(150, minuntasoni);
```

Tämän jälkeen voit käyttää uutta oliota esimerkiksi lukemalla sen sisään tiedoston: aseta arvo attribuuttiin

```
document.uusitaso.src
```

tai kutsu metodia

```
document.uusitaso.load
```

Voit myös kirjoittaa tasoon HTML-koodia käyttämällä tason **document**-attribuuttia:

```
document.uusitaso.document.write("<p>Hei");
```

Kirjoittamisen jälkeen taso täytyy sulkea metodilla:

```
document.uusitaso.document.close();
```

Muista, että voit luoda uusia tasoja ja kirjoittaa tasoihin vasta sitten, kun sivu on latautunut kokonaisuudessaan.

Tasossa voidaan käyttää seuraavia tapahtumankäsittelijöitä: **onMouseOver**, **onMouseOut**, **onLoad**, **onFocus** ja **onBlur**. Niiden määrittelyt lisätään **layer**-komentoon.

LISÄTIETOJA

Lisätietoja JavaScriptistä, siinä käytössä olevista komennoista ja oliosta saat JavaScript-ohjesivulta osoitteesta:

```
http://home.netscape.com/eng/mozilla/3.0/
handbook/javascript/index.html
```

Netscapen versiossa 4 on käytössä uusia JavaScript-ominaisuuksia, joista voit lukea sivulta:

```
http://developer.netscape.com/library/
documentation/communicator/jsguide/js1_2.htm
```

CGI-OHJELMAT

CGI-ohjelmat ovat WWW-palvelimelle sijoitettuja ohjelmia, jotka tulostavat WWW-sivuja. Ne voivat käyttää hyväkseen lomakkeilla tai osoitteen mukana tulevia muuttujia. Ne voivat myös käyttää hyväkseen palvelinkoneella olevia tiedostoja ja muokata niitä.

CGI-ohjelmien tekemiseen tarvitaan palvelimen ylläpitäjän tekemä CGI-hakemisto. Voit kysyä omalta Internet-operaattoriltasi, onko sinun tunnuksellasi mahdollista tehdä CGI-ohjelmia. Jos sinulla on käytössäsi oma WWW-palvelin, voit tehdä sinne CGI-ohjelmia halusi mukaan.

CGI-ohjelmat sijoitetaan palvelinkoneella tiettyyn hakemistoon, joka on määritelty palvelinohjelmiston asetuksissa ohjelmia sisältäväksi hakemistoksi. Tälle hakemistolle määritellään oma URL-osoite, jolloin hakemistossa olevia ohjelmia voidaan ajaa WWW-selaimella.

Esimerkiksi Unix-ympäristössä ajettavassa WWW-palvelimessa voidaan määritellä, että hakemisto nimeltään **`/usr/httpd/cgi-bin`** sisältää CGI-ohjelmia. Palvelimessa määritellään myös, että näitä ohjelmia voidaan ajaa osoitteen **`http://palvelin/cgi-bin/`** kautta (esimerkissä WWW-palvelimen osoite on pelkästään "palvelin" - oikeassa käytössä tämän tilalle sijoitetaan palvelimen varsinainen nimi).

CGI-hakemisto sijaitsee erillään HTML-tiedostoja sisältävistä hakemistoista, joten käyttäjät eivät voi nähdä ohjelmien sisältöä vaan pelkästään niiden ajamisen tuloksena syntyvää tekstiä.

Tämän jälkeen CGI-hakemistoon voidaan sijoittaa ajettavia ohjelmia. Jos hakemistoon on sijoitettu esimerkiksi ohjelma nimeltään **`date`**, se voidaan ajaa osoitteella **`http://palvelin/cgi-bin/date`**. Kun palvelimelta haetaan tässä osoitteessa sijaitsevaa tiedostoa, se huomaa osoitteesta että kyseessä ei ole HTML-tiedosto, vaan ajettava ohjelma. Palvelin käynnistää ohjelman ja ohjaa ohjelman tulosteet WWW-selaimelle.

CGI-ohjelmat voidaan tehdä millä tahansa ohjelmointikielellä. Jotta ohjelma voisi toimia CGI-ohjelmana, sen täytyy tulostaa ensimmäiseksi rivi, jossa kerrotaan minkä tyyppistä tietoa myöhemmin ohjelmassa tulostettava tieto on. Tämän jälkeen ohjelman tulee tulostaa yksi tyhjä rivi. Näiden kah-

den rivin jälkeen tulostuvat tiedot sisältävät varsinaista WWW-selaimelle tarkoitettua dataa, joka ohjataankin suoraan selaimen käsiteltäväksi.

Tehdään esimerkkiohjelma, joka tulostaa käyttäjälle ajohetken päivämäärän ja kellonajan. Ohjelma voidaan tehdä millä kielellä tahansa. Yksi suosituimpia CGI-ohjelmien tekemiseen tarkoitettuja ohjelmointikieliä on **Perl**, jota käytetään alla olevissa esimerkeissä.

Ohjelma, joka tulostaa kellonajan näyttää **Perl**-kielellä ohjelmoituna seuraavanlaiselta:

```
#!/usr/local/bin/perl
# Tulostaa kellonajan
print "Content-type: text/html\n\n";
$aika=time;
($ss, $mm, $tt, $pp, $kk, $vv, $vkp, $yday,
 $isdst)=localtime($aika);
$kk++;
$minuutit=sprintf("%02d", $mm);
$viikonpaiva=(Su,Ma,Ti,Ke,To,Pe,La,Su)[$vkp];
print "<h1>Kellonaika</h1>\n";
print "Nyt on $viikonpaiva $pp.$kk.$vv. \n";
print "Kello on $tt:$minuutit\n";
```

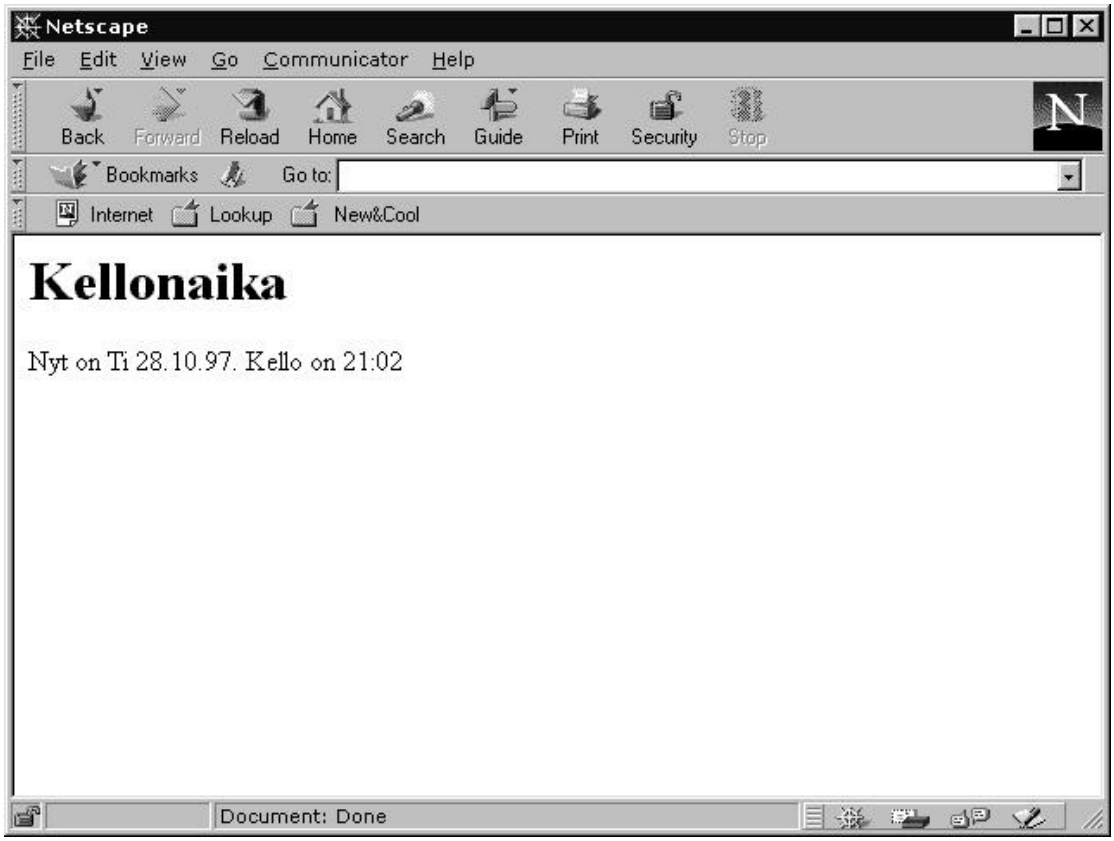
Ohjelman ensimmäinen rivi on Unix-ohjelmien tapa kertoa, että kyseessä on **Perl**-ohjelma. Toisella rivillä on kommentti, joka kertoo ohjelman toiminnasta. Seuraavalla rivillä tulostetaan CGI-ohjelmille yhteinen aloitusrivi, joka kertoo selaimelle, että ohjelma on aikeissa tulostaa HTML-koodia. Tämän jälkeen tulostetaan tyhjä rivi ja siirrytään varsinaisen ohjelman pariin.

Neljännellä rivillä otetaan menossa oleva aika talteen muuttujaan **$aika**. Aikatieto on **Perlin** sisäisessä muodossa, joten se puretaan kahdella seuraavalla rivillä muuttujiin, jotka sisältävät tunnit, minuutit, päivän, kuukauden ja vuoden numeron sekä viikonpäivän.

Kuukausi on alunperin muodossa 0-11, joten seuraavalla rivillä se muutetaan tavallisempaan muotoon 1-12. Tämän jälkeen minuuttimäärä kopioidaan muuttujaan **$minuutit** siten, että lukumäärä on aina kaksinumeroinen (esimerkiksi 01, 30 tai 59). Tämän jälkeen muuttujaan **$viikonpaiva** sijoitetaan viikon päivä suomenkielisenä. Muuttuja **$vkp** sisältää viikonpäivän numerona (0=Sunnuntai, 7=Lauantai).

Viimeiset 3 riviä tulostavat HTML-tekstiä, jonka seassa on muuttujia. Muuttujat muutetaan tulostushetkellä tekstiksi, joka tulee näkyviin käyttäjän selaimessa.

Ohjelmaa kutsutaan osoitteella **`http://palvelin/cgi-bin/date`**. WWW-palvelin ajaa ohjelman ja siirtää ohjelman tulostamat tekstit (**`Content-type`** -riviä ja tyhjää riviä lukuunottamatta) selaimen käsiteltäväksi. Lopputulos näyttää selaimessa tältä:



CGI-ohjelmat voivat tulostaa muutakin kuin HTML-tekstiä. Ensimmäinen ohjelman tulostama rivi määrittelee sen, miten selain käsittelee ohjelman tulostaman tiedon. CGI-ohjelmalla voi tehdä esimerkiksi **gif**-kuvia. Silloin ohjelma tulostaisi alussa seuraavanlaisen rivin:

```
Content-type: image/gif
```

Tämän rivin jälkeen tulostettaisiin yksi tyhjä rivi ja sitten varsinainen **gif**-kuva binäärimuodossa.

CGI-ohjelmat voivat saada parametrejä lomakkeiden tai osoitteen välityksellä. **`Date`**-ohjelmaa voitaisiin laajentaa siten, että sille voitaisiin antaa

osoitteen yhteydessä parametrinä aikavyöhykkeen nimi ja se tulostaisi halutun aikavyöhykkeen ajan. Parametri voidaan antaa osoitteen yhteydessä:

```
http://palvelin/cgi-bin/date?EET
```

CGI-ohjelma voi lukea osoitteen yhteydessä annetun parametrin komentoriviparametrinä, esimerkiksi **Perl**-ohjelmassa komennolla:

```
$aikavyohyke=@ARGV[0];
```

LOMAKKEIDEN KÄSITTELY

Aiemmissa luvuissa tehtiin lomakkeita, joiden tiedot lähetettiin eteenpäin sähköpostin välityksellä. Lomakkeita voidaan käsitellä kuitenkin paljon monipuolisemminkin.

Lomakkeille määriteltiin aina jokin osoite, joka käsittelee lomakkeella annetut tiedot. Lomakkeen aloituskomennon **action**-attribuutissa voidaan antaa osoite CGI-ohjelmalle, joka ottaa vastaan lomakkeelta annetut tiedot ja käsittelee niitä edelleen.

Lomakkeita käsittelevä ohjelma voidaan tehdä esimerkiksi **Perl**-kielellä. Tarvitset käsittelyohjelmien tekemiseen oikeudet sijoittaa CGI-ohjelmia WWW-palvelimesi hakemistoihin, aivan kuten edellisen luvun esimerkeissä.

Lomakkeelta **POST**-muodossa lähetetyt tiedot voidaan purkaa **Perlin** sisäisiin muuttujiin seuraavanlaisella ohjelmanpätkällä:

```
sub readform {
# Lukee sisään POST-lomakkeen tiedot ja
# sijoittelee ne muuttujiin
# sisäisiä muuttujia


my ($buffer, @pairs, $pair, $value, $name);
read(STDIN, $buffer, $ENV{'CONTENT_LENGTH'});
@pairs = split(/&/, $buffer);
  foreach $pair (@pairs) {
       ($name, $value) = split(/=/, $pair);
        $value =~ tr/+/ /;
        $value =~ s/%([a-fA-F0-9]
[a-fA-F0-9])/pack("C", hex($1))/eg;
        $name =~ tr/+/ /;
        $name =~ s/%([a-fA-F0-9]
[a-fA-F0-9])/pack("C", hex($1))/eg;
        $form{$name} = $value;
    }
}
```

Muuttujia voidaan käyttää käyttämällä assosiatiivista muuttujaa **$form** ja antamalla sille indeksiksi lomakkeen osan nimen. Esimerkiksi lomakkeen tekstilaatikossa "sukunimi" annettu nimi voidaan sijoittaa muuttujaan seuraavalla tavalla:

```
$sukunimi=$form{'sukunimi'};
```

GET-muodossa lähetettyjä lomakkeita voidaan lukea samalla ohjelmanpätkällä - ainoastaan rivi, joka alkaa sanalla **read** tulee vaihtaa seuraavaan muotoon:

```
$buffer = $ENV{'QUERY_STRING'};
```

Kun **Perl**-ohjelma on käsitellyt lomakkeelta saamansa tiedot, se voi tulostaa HTML-kielisen sivun. Tämä sivu tulee näkyviin lomakkeen lähettäjälle uutena sivuna.

Alla on esimerkki ohjelmasta, joka ottaa lomakkeelta annetut tiedot ja sijoittaa ne tiedostoon. Tiedoston nimi on määritelty ohjelman alkuriveillä. Aina kun joku lukijoista lähettää lomakkeen, sen tiedot sijoitetaan tiedoston loppuun aiemmin lähetettyjen tietojen perään. Lomakkeen kentät erotetaan tiedostossa toisistaan tabulaattorimerkillä ja ne tallennetaan kenttien nimien perusteella aakkosjärjestyksessä.

```
#!/usr/local/bin/perl
# Ottaa vastaan lomakkeen tiedot
# ja sijoittaa ne tiedostoon.
# Lomakkeen täytyy käyttää
# POST-metodia!

# Tässä määritellään tiedosto, jonne
# tiedot sijoitetaan. CGI-ohjelmalla
# täytyy olla kirjoitusoikeudet
# tiedostoon.
$tiedosto="/usr/home/lomaketulokset.txt";

sub readform {
# Aliohjelma
# Lukee sisään POST-lomakkeen tiedot ja
# sijoittelee ne muuttujiin
# sisäisiä muuttujia
```

```
my ($buffer, @pairs, $pair, $value, $name);
read(STDIN, $buffer, $ENV{'CONTENT_LENGTH'});
@pairs = split(/&/, $buffer);
  foreach $pair (@pairs) {
       ($name, $value) = split(/=/, $pair);
        $value =~ tr/+/ /;
        $value =~ s/%([a-fA-F0-9]
[a-fA-F0-9])/pack("C", hex($1))/eg;
        $name =~ tr/+/ /;
        $name =~ s/%([a-fA-F0-9]
[a-fA-F0-9])/pack("C", hex($1))/eg;
        $form{$name} = $value;
    }
}

# Varsinainen ohjelma alkaa...

# Kirjoitetaan lomakkeelta tulleet
# tiedot tiedostoon.
&readform;
open(OUT, ">>$tiedosto") || die "En voi kirjoittaa
tiedostoon nimeltään $tiedosto, koska $!\n";
foreach $i (keys %form) {
  print OUT $form{$i};
}
print OUT "\n";
close (OUT);

# Lopuksi tulostetaan kiitossivu
# käyttäjälle.
print "Content-type: text/html\n\n";
print "<h1>Kiitos!</h1>\n";
print "<p>Antamasi tiedot on tallennettu. ";
print "Voit nyt peruuttaa taaksepäin ";
print "selaimesi Back-painikkeella.\n";
```

WWW-SIVUJEN FILOSOFIAA

Suurten sivukokonaisuuksien suunnittelu tuo eteen uusia haasteita. Tässä luvussa tutustutaan niihin ja paneudutaan syvemmälle sivujen suunnitteluun.

TEHOKKAIDEN JA HYVIEN SIVUJEN SUUNNITTELU

Sivujen suunnittelu sisältää monia asioita, joita ei välttämättä aluksi osaa ottaa huomioon. Seuraavissa luvuissa käsitellään näitä asioita. Lisäksi annetaan vinkkejä siitä, kuinka verkkosivuille tulee kirjoittaa.

SUUNNITTELE ULKOASU ENNALTA

WWW-sivujen tekemiseen ei kannata rynnätä suinpäin, etenkään jos olet tekemässä suurehkoa sivukokonaisuutta. Sen sijaan ulkoasun ja rakenteen suunnitteluun kannattaa varata runsaasti aikaa. Huomaat suunnittelun tärkeyden etenkin silloin, jos sivujen rakentamisen puolivälissä huomaatkin jättäneesi sivuilta pois oleellisia ominaisuuksia ja joudut rakentamaan ulkoasun uudelleen.

Jos olet tekemässä WWW-sivuja tilaustyönä ja asiakkaasi haluaa nähdä valmiin lopputuloksen mahdollisimman nopeasti, rakenna sivuistasi "pienoismalli". Voit piirtää kuvankäsittelyohjelmaa käyttäen mallikuvan siitä, miltä lopulliset sivut näyttävät. Jos asiakkaasi haluaa muutoksia sivujen ulkoasuun, voit tehdä halutut muutokset nopeasti kuvankäsittelyohjelmalla.

Kun siirryt lopullisten sivujen toteuttamiseen, voit leikata mallikuvastasi kuvat **gif**-tiedostoiksi lopullisilla sivuillasi käytettäväksi.

KÄYTÄ TYYLITIEDOSTOJA

Tyylitiedostojen käyttö helpottaa sivujesi suunnittelua huomattavasti. Kun määrittelet mahdollisimman monet sivujen ominaisuudet (värit, fonttikoot, taustakuvat) erillisissä tyylitiedostoissa olevilla tyylimäärittelyillä ja kytket tyylitiedoston kaikkiin samaa ulkoasua käyttäviin tiedostoihin, voit muuttaa kaikkien sivujen ulkoasua yhtä tyylitiedostoa muokkaamalla.

Jos haluat käyttää jollain sivulla yhteisestä tyylitiedostosta poikkeavia tyylimäärittelyitä, voit lisätä nämä määrittelyt kyseisen sivun alkuun tai itse komentoihin.

Voit myös yhdistää tyylitiedostoja **`@import`** -komennolla. Tällä tavalla voit määritellä tietyille asiakokonaisuuksille erillisiä tyylitiedostoja, jotka ensin perivät kaikille sivuille yhteiset tyylit ja muokkaavat sitten haluttuja kohtia kyseisen asiakokonaisuuden ulkoasun mukaiseksi.

KUINKA KIRJOITETAAN WWW:TÄ VARTEN?

WWW:hen kirjoittaminen eroaa jonkin verran muusta kirjoittamisesta. Käyttäjäsi eivät jaksa lukea näytöltä yhtä pitkiä tekstejä kuin paperilta eikä heidän keskittymiskykynsä aina riitä pitkien asioiden seuraamiseen.

Tämä johtuu nykyisistä laitteistoista: tietokoneiden näyttöjen käyttömukavuus on vielä kaukana kirjojen tai lehtien mukavuudesta - tietokoneen ruudulla olevaa tekstiä luetaan työtuolissa istuen, useimmiten pystysuorassa olevalta näytöltä, jonka tarkkuus ei ole paperin veroinen ja jonka valaistus rasittaa aina jonkin verran silmiä. Kun näytöllä näkyvän tekstin määrä saattaa olla vieläpä hyvinkin rajoitettu ja käyttäjä joutuu rullaamaan tekstiä jatkuvasti, on selvää että käyttömukavuus ei vastaa paperin käyttömukavuutta.

Tulevaisuudessa nämäkin asiat saattavat muuttua jatkuvasti parantuvan tekniikan ansiosta, mutta tällä hetkellä WWW:hen kirjoittaessasi joudut mukautumaan tekniikan aiheuttamiin rajoituksiin.

Kun kirjoitat WWW-sivuille tarkoitettua tekstiä, pidä mielessäsi muutama perussääntö: kirjoita lyhyesti, kirjoita silmäiltävissä olevaa tekstiä, jaa tekstisi sopivankokoisiksi annoksiksi ja sijoita tärkeimmät asiat ensimmäisiksi.

KIRJOITA LYHYESTI

Ihmiset lukevat tekstiä tietokoneen ruudulta keskimäärin 25% hitaammin kuin paperilta. Koska tietokoneen ruudulta lukeminen on tekniikan rajoituksista johtuen lisäksi vielä paperilta lukemista epämukavampaa, luettavaa tekstiä tulisi olla vieläkin vähemmän. Peukalosääntönä voisi pitää, että verkkoon kannattaa kirjoittaa puolet tekstistä, jonka kirjoittaisit paperille.

KIRJOITA SILMÄILTÄVÄÄ TEKSTIÄ

Tyypillinen verkon käyttäjä on yleensä jonkin verran kärsimätön. Hän maksaa usein verkkoyhteydestään minuuttimaksua ja haluaa sen vuoksi käyttää jokaisen linjallaolosekunnin hyväksi. Koska verkon luonteesta johtuen palvelimesta toiseen siirtyminen on äärimmäisen helppoa, käyttäjä haluaa koko ajan etsiä uusia kokemuksia ja on sen vuoksi innokas siirtymään eteenpäin. Lisäksi, koska tietokoneen ruudulta lukeminen on lievästi epämukavaa, hän ei halua viettää ylimääräistä aikaa tekstiä lukemalla.

Kaikki nämä tekijät yhteydessä aiheuttavat sen, että käyttäjä mieluimmin silmäilee ruudulla näkyvää tekstiä kuin lukee sitä läpi sana kerrallaan. Hän etsii tekstin seasta kiinnostavia sanoja, lauseita ja kappaleita sekä hyppii mielenkiinnottoman tekstin yli.

Jos haluat, että sivullasi oleva teksti tulee luetuksi edes osittain tai että siinä olevat ideat menevät perille käyttäjälle, sinun tuleekin kirjoittaa tekstistäsi silmäilykelpoista. Silmäilykelpoinen teksti sisältää kohtia, jotka vangitsevat lukijan mielenkiinnon ja ohjaavat hänen katseensa mielenkiintoisiin kohtiin tekstissä.

Käyttämällä paljon väliotsikoita voit tehdä tekstillesi selkeän rakenteen. Kun teet väliotsikot hierarkkisen rakenteen mukaisiksi, voit helposti näyttää tekstistä kohtia, jotka sisältävät uutta asiaa edelliseen verrattuna. Hierarkkisissa otsikoissa väliotsikot erottuvat korkeamman tason otsikoista selkeästi ja nopeasti. HTML-kielen otsikkokomennot ovat omiaan näiden otsikoiden tekemiseen - lisäetuna sivusi sisältää otsikkokomentojen ansiosta myös sähköisessä muodossa olevan rakenteen, jota voidaan erilaisilla tietokoneohjelmilla käyttää hyväksi.

Muista myös tehdä väliotsikoista informatiivisia. Koska ne ovat todennäköisimmät sivultasi luetut asiat, kerro tärkeimmät asiat itse otsikossa. Asiaankuulumattomat otsikot, joissa on hauskoja lausahduksia tai iskulauseita jäävät lukijan mieleen, mutta varsinainen sivulla kerrottu asia saattaa jäädä tällöin lukematta.

Voit sijoittaa tekstin sekaan myös erilaisia avainsanoja ja -lauseita, joilla voit houkutella lukijan mielenkiinnon kyseiseen tekstin kohtaan. Merkitse avainkohdat muusta tekstistä erottuvalla tyylillä (esimerkiksi **`<em>`** tai **`<strong>`** -komennoilla tai väreillä) niin ne erottuvat tekstistä. Myös linkit erottuvat automaattisesti tekstistä ja tulevat huomatuksi, koska ne ovat yleensä eriväristä kuin muu teksti ja useimmiten lisäksi vielä alleviivattu.

JAA TEKSTI SOPIVIIN PALOIHIN

Jakamalla tekstin sopiviksi annospaloiksi voit tehdä teksteistäsi lyhyempiä ja helpommin silmäiltäviä. Suositeltava koko yhden WWW-sivun sisältämälle tekstille on noin 200-500 sanaa[5].

Lyhyet kappaleet näyttävät käyttäjän mielestä houkuttelevammilta ja todennäköisyys niiden lukemiselle nousee välittömästi. Tällöin myös tekstin lukemiseen tarvitaan paljon vähemmän rullaamista, jolloin lukemisen miellyttävyys korostuu entisestään.

Voit jakaa tekstin osiksi kahdellakin eri tavalla. Toinen tapa on ottaa pitkä teksti ja jakaa se peräkkäisiksi HTML-sivuiksi, jotka kytketään toisiinsa linkeillä. Tällöin ensimmäisen sivun lukenut käyttäjä voi siirtyä syvemmälle tekstiin seuraamalla linkkiä. Voit lisätä tällaisen tekstin käyttömukavuutta lisäämällä peräkkäisille sivuille pienen navigointiapuvälineen, jossa kerrotaan jokaisen osasivun sisältämät asiat ja annetaan suora linkki tälle sivulle. Näin käyttäjä voi hypätä tylsiltä näyttävien sivujen ohi suoraan itseään kiinnostavaan asiaan (ja kenties palata muille sivuille myöhemmin).

Toinen tapa on käyttää hyväkseen WWW:n hypertekstiominaisuutta. Tällöin voit kirjoittaa tekstisi ensimmäisen sivun yhteenvedon muotoon ja tarjota asiasta kiinnostuneille enemmän lukemista tekstissä olevien avainsanojen kautta tehtyjen linkkien avulla. Käyttäjä voi silmäillä sivun nopeasti läpi ja jos asia ei vaikuta kiinnostavalta, hän voi hyvillä mielin siirtyä eteenpäin seuraavaan aiheeseen. Jos hän kiinnostuu aiheesta, hän voi avainkohtia napsauttelemalla lukea lisätietoja samasta aiheesta. Tällöin teksti on jaettu hierarkkiseksi kokonaisuudeksi. Tällaisessakin tapauksessa tekstikokonaisuudesta kannattaa tehdä näkyville jonkinnäköinen sisällysluettelo, jotta käyttäjä voi siitä tutkia sisältääkö teksti häntä kiinnostavaa asiaa ja hypätä halutessaan siihen suoraan.

Kumpaa tahansa tapaa käytätkin, jaa teksti kuitenkin paloiksi aina kokonaisuuksittain. Sivulta toiselle siirtyminen tekee aina jonkin mittaisen tauon lukemiseen, joka katkaisee käyttäjän keskittymisen. Jos tämä tapahtuu kesken ajatusrakenteen, käyttäjän mielenkiinto asiaan herpaantuu aina jonkin verran.

---

[5] **http://www.cnet.com/Content/Builder/Graphics/Design/ss2b.html**

Kokonaisuuksittain paloiksi jakaminen helpottaa myös tulostamista: jos käyttäjä pitää tekstistä niin paljon, että hän haluaa tulostaa siitä itselleen oman kopion, hän voi tulostaa välistä haluamansa asiakokonaisuudet. Jos oletat että tekstilläsi on laajempaakin mielenkiintoa, harkitse myös kokonaisen tekstin tarjoamista yhdellä sivulla niitä varten, jotka haluavat printata koko asian yhdellä kerralla.

KIRJOITA TÄRKEIMMÄT ASIAT ENSIMMÄISIKSI

Yksi WWW:n käyttäjiä yhdistävä seikka on se, että he eivät pidä tekstin vierittämisestä. Kun käyttäjä saapuu WWW-sivulle, sivun yläosa on kohta, minkä kaikki sivulle saapuvat näkevät. Loppuosa sivusta näkyy vain niille käyttäjille, jotka vierittävät sivua alaspäin.

Sivun vierittäminen vaatii useimmiten pienen fyysisen ponnistuksen: käyttäjän täytyy joko irrottaa otteensa hiirestä ja painaa välilyönti- tai **Page Down** -näppäintä tai siirtää hiirensä kursoria vierityspalkin kohdalle ja painaa nappia. Jos sivu ei vaikuta tarpeeksi mielenkiintoiselta, useat käyttäjät eivät vaivaudu tekemään tätä.

Tämän vuoksi on tärkeää, että sijoitat sivun tärkeimmän asian yläreunaan. Voit sijoittaa tänne provosoivan tekstikappaleen, väitteen tai muunlaisen tekstin, joka herättää käyttäjän mielenkiinnon. Jos tarjoat linkkejä eteenpäin, voit sijoittaa linkin yhteyteen näkyville otteen seuraavan sivun mielenkiintoisimmasta tarjonnasta.

Alempana sivulla (tai alasivuilla) voit käsitellä samaa asiaa laajemmin ja perusteellisemmin - mutta muista houkutella lukijaa niin, että hänellä on mahdollisuus päästä tälle alasivulle.

Voit houkutella lukijaa syvemmälle myös kuvilla ja muilla tehosteilla, mutta pidä mielessä, että kuvien lataaminen hidastaa myös sivun lataamista ja saattaa jo sen vuoksi karkottaa käyttäjän pois.

KYSYMYKSIÄ SIVUJEN SUUNNITTELIJOILLE

Sivuja suunnitellessa sinun kannattaa kysyä itseltäsi myös muutamia kysymyksiä, joiden avulla voit tarkentaa itsellesi, miltä sivujen tulisi näyttää ja miten niiden toimivuus ja ulkoasu tulisi toteuttaa. Mitä suuremmalle organisaatiolle suunnittelet sivujasi, sitä tärkeämpää huolellinen suunnittelu on.

MIKSI HALUAT TEHDÄ WWW-SIVUT?

Kysy itseltäsi, miksi ja mihin tarkoitukseen sivuja tehdään.

Onko tarkoituksena tehdä sivut, jotka toimivat WWW-käyntikorttina? Internetin alkuaikoina monilla yrityksillä oli tärkeimpänä tavoitteena saada aikaan pelkkä näkyvyys verkossa - tällä tavalla yritys antoi itsestään kuvan aikaansa seuraavana yrityksenä, jolle Internet ei ole vieras asia.

Käyntikorttisivujen tekemisessä ei ole välttämättä mitään hävettävää, mutta mieti myös voisitko tehdä sivut, joista on oikeaa lisäarvoa.

WWW-sivuilla voit julkaista esimerkiksi sellaista tietoa, jonka julkistaminen muissa medioissa saattaisi olla epäkäytännöllistä. Voit tarjota runsaasti taustatietoa ja yksityiskohtaisia tuote-esittelyitä. WWW-sivujen interaktiivisen luonteen ansiosta voit jakaa tiedon siten, että jokainen lukijasi pystyy löytämään juuri itseään kiinnostavat aiheet ja ohittamaan sellaisen tiedon, jolla ei ole merkitystä hänelle.

KENELLE TEET SIVUT?

Mieti kohderyhmääsi! Suunnitteluvaiheessa saat sivujen sisältöön liittyviä toiveita ja ehdotuksia lähinnä oman organisaatiosi sisältä. Nämä toiveet saattavat erota kuitenkin siitä, mitä yleisösi haluaa nähdä.

Yritä asettua potentiaalisen käyttäjän asemaan ja miettiä, mitä tavallinen verkon käyttäjä haluaa löytää sivuiltasi sinne saapuessaan. Onko organisaatiollasi tarjolla jotain sellaista tietoa, joka olisi omiaan verkossa julkaistavaksi?

Mieti tarjolla olevan tiedon tärkeysjärjestys ja nosta sivuillasi näkyvimmälle paikalle sellainen tieto, jonka arvelet kiinnostavan sivuillesi saapuvaa lukijaa.

PIDETÄÄNKÖ SIVUJA YLLÄ?

Pelkkä sivujen suunnittelu ja valmistaminen ei riitä toimivan WWW-palvelun rakentamiseen vaan siihen kuuluu olennaisena osana myös ylläpidon suunnittelu.

Suurien WWW-palveluiden ylläpito ja niihin lisättävien päivitystietojen suunnittelu on tärkeä osa palvelun toimintaa. Muista varmistaa ylläpidon toimivuus määrittelemällä tarpeeksi resursseja WWW-palvelun ylläpitoon.

Yleinen virhe on pystyttää palvelu suurella rahalla mutta jättää ylläpito työntekijöille, jotka hoitelevat palvelinta oman työn ohessa. Isoissa organisaatioissa WWW-palvelun ylläpitoon onkin syytä asettaa työntekijöitä, jotka voivat keskittyä siihen täyspäiväisesti.

WWW-sivut tarvitsevat ylläpitoa monellakin tavalla: uuden julkistettavan tiedon siirtäminen sivuille, grafiikoiden päivittäminen, linkkien tarkistus, palautteen käsittely ja palvelun ennemmin tai myöhemmin tapahtuvan laajentamisen suunnittelu ovat osa monista ylläpidossa vastaan tulevista asioista.

Jakob Nielsenin mukaan ylläpitoon kannattaa budjetoida resursseja saman verran tai vähintään puolet palvelun rakentamiseen käytetyistä resursseista[6].

ONKO WWW TOISSIJAINEN JULKAISUMUOTO?

Monien yrityksien WWW-sivut sisältävät jo aikaisemmin (esimerkiksi paperimuodossa) julkaistua materiaalia HTML-muotoon siirrettynä. Tämä on WWW:n mahdollisuuksien haaskausta, sillä muusta muodosta muunnettu materiaali ei välttämättä käytä kaikkia verkon mahdollisuuksia hyväkseen.

Aloittelevan WWW-palvelun pystyttämisessä jo olemassa olevan materiaalin hyväksi käyttäminen on hyvä idea, mutta jatkossa kannattaa harkita uuden materiaalin tuottamisen siirtämistä verkkoon. Kun erilainen informaatio julkaistaan ensimmäisenä verkossa ja siitä tehdään muissa medioissa julkaistavia versioita, verkossa olevaa materiaalia voidaan käyttää aina lähteenä muita versioita tehtäessä.

Verkossa olevaa tietoa voidaan päivittää välittömästi tiedon muuttuessa. Tällä tavalla esimerkiksi paperiesitteiden suunnittelijat voivat tarkistaa WWW-sivuilta viimeisimmät tiedot ja käyttää niitä tehdessään paperilla julkaistavaa materiaalia.

---

[6] **http://www.useit.com/alertbox/9706b.html**

OVATKO INTRA JA INTER ERI ASIA?

Jos käytät verkkoa organisaation sisäiseen julkaisemiseen (intranet-käyttöön), tee sisäisille käyttäjille eri aloitussivu. Suurelle yleisölle tarkoitettu aloitussivu voi sisältää tervetulosivuja, houkuttelevaa grafiikkaa ja mainoksia, mutta älä pakota henkilökuntaasi katselemaan näitä.

Koska intranetin käyttäjät ovat jo tuttuja tarjoamiesi palveluiden kanssa etkä yritä myydä heille tuotteitasi, voit tehdä sisäiseen käyttöön aloitussivun, jonka avulla paikalliset käyttäjät pääsevät nopeasti kiinni haluamaansa tietoon. Intranet-etusivun tärkeänä ominaisuutena on myös nopea latautuminen, joten voit jättää siltä sivulta pois ylimääräiset kuvat ja koristeet.

LIITTEET

SANASTO

AIFF Ääniformaatti.

Aloituskomento Jotkut HTML-komennot koostuvat aloitus- ja lopetuskomennosta. Komennon vaikutus ulottuu tällöin näiden komentojen väliin jäävälle alueelle.

Appletti (**Applet**) Java-kielellä rakennettu ohjelma, joka ladataan lukijan koneeseen ja ajetaan siellä. Ohjelman tulosteet näkyvät sivun sisällä ikään kuin interaktiivisena kuvana.

Apuohjelma (**External Viewer**) Ohjelma, joka näyttää tiedostot, joita selain ei osaa näyttää. Apuohjelma voi esimerkiksi soittaa äänitiedostoja. Tarkoitukseen voidaan käyttää myös **plug-in**-ohjelmia.

ASCII Merkistöstandardi. Lähes kaikki nykyiset tietokoneet käyttävät merkistössään ASCII-standardia. Termi ASCII-teksti tarkoittaa tekstiä, jossa ei ole muotoilukomentoja. Tekstinkäsittelyohjelmalla kirjoitetut tekstit eivät ole puhdasta ASCII-tekstiä ennen kuin ne on tallennettu siinä muodossa. Tekstinkäsittelyohjelmalla tehdyt dokumentit voi muuttaa ASCII-tekstiksi tallentamalla ne "**text only**" -muodossa.

Asiakasohjelma (**Client**) Ohjelma, joka toimii yhdessä palvelinkoneen kanssa. Asiakasohjelma itsessään ei tuota tietoa, vaan sen pääasiallinen tehtävä on ottaa yhteyttä palvelimiin ja pyytää niiltä tietoja ja esittää sitten tämä tieto käyttäjälle. Käyttämäsi selain on esimerkki asiakasohjelmasta.

Attribuutti (olioissa) (Attribute) Olio voi sisältää useita muuttujia, attribuutteja.

Attribuutti HTML-komennon osa, joka tarkentaa komennon toimintaa. Esimerkiksi komento **`<select>`**

aloittaa valikon ja komento **`<select multiple>`** aloittaa valikon, josta voi valita monta kohtaa. Esimerkissä sana **`multiple`** oli attribuutti.

AU Ääniformaatti.

AVI Videoleikeformaatti.

Boottaaminen Tietokoneen käynnistäminen uudelleen.

CGI (**Common Gateway Interface**) Palvelimien standardi, joka määrittelee CGI-ohjelmien saamat tiedot sekä tavan, jolla ohjelmat tulostavat tietoa.

CGI-ohjelma CGI-ohjelmat sijaitsevat palvelimen ohjelmahakemistossa. Niitä kutsutaan selaimesta käyttäen ohjelmaan osoittavaa URL-osoitetta. Saatuaan CGI-ohjelmaan osoittavan hakupyynnön palvelin käynnistää ohjelman ja lähettää ohjelman tulosteet selaimelle. CGI-ohjelmien avulla on mahdollista toteuttaa lomakkeilta syötettyjä tietoja käsitteleviä ohjelmia, sivuja, joiden sisältö muuttuu kellonajan tai lukijan sijainnin mukaan, hakuohjelmia ja niin edelleen.

Entiteetti HTML:n tapa esittää merkki, joka ei kuulu normaaliin merkistöön. Esimerkiksi Ä-kirjainta vastaava entiteetti on **`Ä`**

Extranet Rajoitettu Internet-palvelu. Yritys voi sijoittaa vain tietyille asiakkaille tarkoitettua tietoa Extranet-palvelimeen ja suojata tämän palvelimen salasanalla.

Fetch Tiedostonsiirto-ohjelma Macintoshille.

Flash Formaatti multimediaohjelmille.

Formaatti Tallennustapa tai -muoto. Esimerkiksi kuva saattaa olla tallennettuna joko **gif**- tai **jpeg**-formaatissa. Jos kuva on tallennettu käyttäen

gif-formaattia, se vie enemmän tilaa kuin **jpeg**-formaatissa tallennettu kuva. **Jpeg**-formaatissa tallennetusta kuvasta sen sijaan katoavat pienet yksityiskohdat.

Ftp (**File Transfer Protocol**) Protokolla jota käytetään tiedostojen siirtämiseen Internetissä. Myös ohjelma, jolla voi siirtää tiedostoja tiedostoarkistoihin ja niistä itselle.

Funktio (**Function**) Ohjelmassa oleva kokoelma komentoja, jotka on yhdistetty yhden nimen alle. Tällä tavalla ohjelmanpätkää voidaan käyttää useammassa paikassa ilman, että sen koodia täytyy kirjoittaa uudelleen.

Fyysinen (Fyysinen muotoilukomento) Kertoo miltä jonkin tulee näyttää lukijan kuvaruudulla. Loogisen "vastakohta". Esimerkiksi lihavointikomento on fyysinen komento.

Gopher WWW:n "edeltäjä". Valikkopohjainen järjestelmä, jonka avulla pystyi tarjoamaan tekstejä luettavaksi kaikille verkon käyttäjille.

Hakuohjelma (Robotti, mato) Ohjelma, joka selaa rekursiivisesti (seuraa kaikkia löytämiään linkkejä) WWW:n sivuja ja muodostaa niistä hakutietokannan. WWW:n käyttäjät voivat käyttää tätä tietokantaa hyväkseen etsiessään tietoa WWW:stä.

Heksadesimaali Heksadesimaalijärjestelmässä (eli 16-kantajärjestelmässä) käytetään kymmenen numeron sijasta kuuttatoista numeroa. Tavallisessa lukujärjestelmässä käytössä ovat numerot nollasta yhdeksään. Heksadesimaaliluvut sisältävät lisäksi kirjaimet **A**-**F**. Niiden avulla on mahdollista esittää kahdella numerolla (/kirjaimella) luvut 0-255. Numero 9 on heksadesimaalisena **9**, numero 10 on heksadesimaalisena **A**. Numero 255 heksadesimaalisena on **FF**.

HTML (**Hypertext Markup Language**) Merkintäkieli, jonka avulla tehdään WWW-sivuja. Tällä kielellä merkitään tekstiin linkit, erilaiset tekstin korostukset, kuvitus jne.

HTML-komento Komentosana, joka alkaa merkillä "**<**" ja loppuu merkkiin "**>**". Välissä oleva komento antaa selaimelle ohjeita tekstin muotoilusta. HTML-komento voi sisältää komennon lisäksi attribuutteja. Esimerkki: **<img src="kuva.gif">** tai **<p>**.

HTML-tiedosto Tekstitiedosto, jossa on HTML-komentoja. Kun selain lataa HTML-tiedoston, se muotoilee tekstin selaimen ikkunaan WWW-sivuksi.

Http WWW-sivujen siirtoon käytettävä protokolla.

Hypermedia Sisältää hypertekstin lisäksi kuvia, ääntä, liikkuvaa kuvaa jne.

Hyperteksti Teksti, joka sisältää linkkejä toisiin teksteihin.

Interaktiivinen kuva (Klikattava kuva) Kuva, joka sisältää kuumia alueita. Lukija voi näpäyttää hiirellä jotain kohtaa kuvasta, jolloin hän saa näkyviin uuden sivun.

Interlace **Interlace**-muodossa tallennettu **gif**-kuva tulee ruudulle näkyviin ensin epätarkkana, sitten yhä tarkempana kunnes se näkyy lopullisessa muodossaan. **Interlace**-muodossa tallennetut kuvat eivät lataudu sen nopeammin kuin tavallisetkaan kuvat, mutta niiden sisällöstä saa nopeammin käsityksen.

Internet Explorer Microsoft Windows '95:n mukana tuleva ilmainen selain. Saatavilla myös vanhempaan Windowsiin ja Macintoshiin.

Internet-operaattori Katso "operaattori".

Intranet Rajoitettu Internet-palvelu. Esimerkiksi yritys voi sijoittaa vain henkilökunnalle tarkoitettua tietoa Intranet-WWW-palvelimeen. Palvelin suojataan salasanalla siten, että vain työntekijät voivat lukea siellä olevia sivuja. Intranet voi sisältää myös muitakin Internet-palveluita, kuten sähköpostin, tiedostojen siirtomahdollisuuden jne. Kaikki palvelut on kuitenkin rajoitettu vain niille, joilla on Intranetin käyttöoikeus.

ISO-8859-1 Merkistöstandardi. Suomalaisille käyttäjille standardilla on merkitystä lähinnä Å-, Ä- ja Ö-kirjaimien esityksessä. WWW käyttää tätä merkistöstandardia, samoin kuin Microsoft Windows, Commodore Amiga ja 8-bittiset Unix-järjestelmät. Muista laitteistoista tai järjestelmistä tuleva teksti ei välttämättä sisällä oikeanlaisia ääkkösmerkkejä.

Java Ohjelmointikieli, jolla voi esimerkiksi toteuttaa pieniä ohjelmia ladattavaksi WWW-sivujen sisään. Java-kielellä kirjoitetut ohjelmat käännetään ja linkitetään HTML-tiedostoon **`applet`**-komennolla.

JavaScript Javan "kevyempi versio". JavaScriptillä toteutetut ohjelmat kirjoitetaan lähdekoodina HTML-tiedoston sisään, josta selain tulkitsee ne näyttäessään sivua.

Kehys (**Frame**) Uusimmat selaimet sisältävät komentoja, joilla selaimen ikkuna voidaan jakaa useaan osaan, *kehykseen*, siten että jokaisessa kehyksessä on näkyvissä eri sivu.

Kirjanmerkki (**Bookmark**) Kirjanmerkkejä on kahdenlaisia. Selainten kirjanmerkit ovat sivujen osoitteita, jotka selain on tallentanut listaksi myöhempää käyttöä varten. Niiden avulla usein luettuihin sivuihin voi hypätä suoraan. HTML:n kirjanmerkit ovat tekstiin lisättäviä komentoja, joihin voi tehdä linkin. Niiden ansiosta linkki voi osoittaa myös tiedoston keskellä olevaan tekstiin.

Konstruktori (**Constructor**) Funktio, joka rakentaa olion.

Koodi Joukko komentoja, esimerkiksi HTML-koodi.

Kotihakemisto (**Home directory**) Unix-käyttöjärjestelmässä (ja muissakin usean käyttäjän käyttöjärjestelmissä) jokaisella käyttäjällä on oma hakemisto, jonne hän voi sijoittaa henkilökohtaisia tiedostojaan. Tätä paikkaa kutsutaan kotihakemistoksi.

Kotisivu (**Home page**) Termiä "kotisivu" käytetään useasta eri asiasta. Organisaation kotisivu on joko organisaation oman palvelimen etusivu tai organisaation tarjoaman sivukokonaisuuden etusivu (jos samassa palvelimessa sijaitsee usean organisaation tietoja). Kotisivu voi olla myös käyttäjän itsestään tekemä esittely, joka sijaitsee hänen kotihakemistossaan tai muussa tarkoitusta varten varatussa paikassa.

Kuuma teksti (Kuuma sana, kuuma alue) Lukijalle näkyvä linkin osa. Kuuma teksti näkyy ruudulla erivärisenä tai eri tavalla muotoiltuna tekstinä. Muotoilu kertoo lukijalle, että hän voi valita kuuman sanan ja saada näkyviin uuden sivun.

Linkki (**Link**) WWW-sivulla oleva osa, jonka valitsemalla saa näkyviin uuden sivun.

Loki (**Log**, tai tilasto) Tiedosto, johon palvelin kirjaa erilaisia asioita. WWW-palvelin pitää tilastoa saamistaan hakupyynnöistä sekä ajon aikana tulleista virheilmoituksista. Jotkut WWW-palvelimet tilastoivat myös käytettyjä selaimia ja reittejä, joita pitkin palvelimeen tultiin.

Lomake (**Form**) Sivun osa, jossa on tekstilaatikoita, painikkeita ja valikoita. Lukija voi täyttää niihin tietoja ja lähettää ne eteenpäin käsiteltäviksi.

Looginen (Looginen muotoilukomento) Kertoo mitä jokin tarkoittaa, ei sitä miltä se näyttää. Fyysisen

"vastakohta". Esimerkiksi vahva korostus (****) on looginen komento.

Lopetuskomento Jotkut HTML-komennot koostuvat aloitus- ja lopetuskomennosta. Komennon vaikutus ulottuu tällöin näiden komentojen väliin jäävälle alueelle. Esimerkiksi aloituskomennon **** ja lopetuskomennon **** välissä olevat teksti näkyy lihavoituna.

Lukuoikeus Unix ja jotkut muutkin käyttöjärjestelmät sisältävät menetelmän, jolla estetään käyttäjää näkemästä toisen käyttäjän tiedostoja. Tämä määritellään lukuoikeuksilla. Tiedoston voi määritellä siten, että ainoastaan tiedoston omistaja voi lukea ja kirjoittaa siihen. Tiedostolle voi antaa myös lukuoikeudet kaikille käyttäjille, jolloin kuka tahansa voi lukea tiedostoa. WWW-palvelimella täytyy olla lukuoikeudet kaikkiin tiedostoihin, joita sen on tarkoitus lähettää käyttäjille.

Lynx Selain, joka ei tarvitse toimiakseen graafista käyttöliittymää. Lynxin saa muun muassa Unix- ja MS-DOS -käyttöjärjestelmille.

Metodi (**Method**) Olioon voidaan kytkeä toimintoja funktioilla. Näitä kutsutaan metodeiksi.

Mozilla Netscape-selaimen työnimi. Nimeä käytetään vielä nykyäänkin joissain yhteyksissä tarkoittamaan Netscape-selainta.

Muuttuja (**Variable**) Ohjelmoinnissa käytettävä termi. Muuttujiin voidaan tallettaa tekstejä ja lukuja ja niitä voidaan käsitellä edelleen.

Netscape Navigator **Netscape Communications Corporationin** valmistama selain. Ohjelmasta käytetään yleensä nimeä "Netscape". Uudemmista versioista käytetään myös nimitystä **Netscape Communicator**.

Olio (**Object**) Monimutkaisempi muuttuja. Olio voi sisältää useita muuttujia (attribuutteja) ja funktioita (metodeja).

Operaattori Yhtiö, jolta olet ostanut Internet-yhteyden. Operaattori tarjoaa sinulle yhteyden verkkoon ja erilaisia Internet-palveluita kuten sähköpostin ja mahdollisuuden tehdä kotisivuja.

Otsikkoalue (**Header**) WWW-sivussa oleva alue, jonne kirjoitetaan sivun teknisiä tietoja.

Paint Shop Pro Eräs kuvankäsittelyohjelma.

Palautearvo (**Return Value**) Funktiot voivat palauttaa laskennan tuloksia palautearvoina. Tällä tavalla voidaan tehdä esimerkiksi funktio, joka laskee laskutoimituksen ja palauttaa tuloksen sitä kutsuvalle ohjelmalle.

Palvelin (**Server**) Ohjelma, johon asiakasohjelmat ottavat yhteyttä ja pyytävät tiedostoja. Palvelin jakaa verkossa oleville ohjelmille tiedostoja pyyntöjen mukaan. Erilaiset palvelimet tarjoilevat erilaisia tiedostoja: WWW-palvelimet tarjoavat WWW-sivuja, **gopher**-palvelimet tekstiä, tiedostopalvelimet tiedostoja, uutispalvelimet uutisryhmien artikkeleita jne.

Palvelinkone Tietokone, jossa ajetaan palvelinta. Palvelinkoneet ovat yleensä normaalia tehokkaampia koneita, mutta käytännössä melkein mikä tahansa verkkoon kytketty tietokone voi toimia palvelinkoneena.

Parametri Ohjelmalle käynnistyksen yhteydessä annettava lisätieto. Jos ohjelman nimi tarkoittaa "*mitä tehdään*", ohjelmalle annettavat parametrit tarkoittavat "*mille tehdään*".

Perl Ohjelmointikieli, jota käytetään usein CGI-ohjelmia tehtäessä. **Perl** on tulkkaava kieli, eli

sillä tehtyjä ohjelmia ei tarvitse kääntää. Tämän ansiosta **Perl**-ohjelman voi kopioida verkosta itselleen ja käyttää sitä ilman erillistä kääntämistä.

Photoshop Eräs kuvankäsittelyohjelma.

Pikseli (**Pixel**) Tietokoneen kuvaruudulla oleva kuva koostuu pienistä pisteistä, joita kutsutaan pikseleiksi. PC:n kuvaruutu koostuu SVGA-tilassa vaakasuunnassa 800 pikselistä ja pystysuunnasta 600 pikselistä.

Plug-in Ohjelma, joka laajentaa selaimen ominaisuuksia. Kun selaimeen on asennettu esimerkiksi **Quicktime-plug-in**, se osaa näyttää **Quicktime**-formaatissa olevia videoleikkeitä suoraan ikkunassa.

Polku (**Path**) Polku osoittaa tiedoston sijainnin hakemistopuussa. Hakemistonimien väliin sijoitetaan erotinmerkki, joka on yleensä kauttaviiva (**/**). Esimerkiksi: `/home/kalle/txt/file.txt` .

MS-DOSissa ja Windowsissa käytetään erottimena kenoviivaa (\).

Postimerkkikuva Tekstissä oleva pieni kuva, joka on samalla linkki samaan kuvaan isompana. Postimerkkikuvia käytettäessä kuvien siirtoon ei kulu pitkää aikaa ja sivut latautuvat selaimeen nopeasti. Lukija näkee kuvituksen ruudullaan pieninä kuvina. Hänellä on kuitenkin halutessaan mahdollisuus saada näkyviin sama kuva isompana ja tarkempana näpäyttämällä hiirellä pikkukuvaa.

Progressiivinen jpeg (**Progressive jpeg**) Kuten **Interlace**-muodossa oleva **gif**-kuva, mutta tallennettu **jpeg**-muodossa. Tulee ladattaessa näkyviin kuvaruudulle ensin sumeana, sitten tarkempana.

Protokolla (**Protocol**) Tiedon siirtotapa. Selaimen ja palvelimen täytyy puhua samaa "kieltä", jotta ne voi-

vat vaihtaa keskenään tiedostoja. Protokolla on tiedonsiirron kieli. Internetissä on käytössä lukuisia erilaisia protokollia erilaisen tiedon siirtämiseen.

Pääteyhteys (**Telnet**) Merkkipohjainen yhteys tietokoneesta toiseen. Pääteyhteyden avulla toista tietokonetta voi käyttää verkon yli mistä tahansa maailmalta. Pääteyhteys voi olla myös mikrosta keskustietokoneeseen tai keskustietokoneesta toiseen tietokoneeseen. Pääteyhteys käyttää **`telnet`**-protokollaa.

Quicktime Videoleikeformaatti.

Reload Selaimen toiminto, joka lataa saman sivun uudelleen. Kun sivu on haettu ja muotoiltu lukijan kuvaruudulle, selain katkaisee yhteyden palvelimeen. Jos palvelimessa sijaitseva sivu muuttuu tai sitä päivitetään, muutokset tulevat näkyviin painamalla **Reload**-painiketta. Tällöin selain avaa uuden yhteyden palvelimeen ja lataa ruudulla näkyvän sivun uudelleen.

Resoluutio Kuvaruudun tai tulosteen tarkkuus. Tarkkuus ilmoitetaan yleensä pikseleinä tuumaa kohti, esimerkiksi 300 pikseliä tuumalla. Kuvaruudun resoluutio voidaan ilmaista myös pikselikokona — esimerkiksi 800x600 pikseliä (800 vaakasuunnassa ja 600 pystysuunnassa).

RGB-arvo Tapa esittää väri kolmen numeron avulla. Väristä annetaan kolmen värikomponentin arvot: punaisen (**Red**), vihreän (**Green**) ja sinisen (**Blue**). Kirjaimet **`RGB`** tulevat värien nimien alkukirjaimista. Mikä tahansa väri voidaan esittää näiden kolmen värin avulla. Esimerkiksi violetti on väri, joka sisältää yhtä paljon punaista ja sinistä, mutta ei vihreää. Harmaa sisältää hieman kaikkia kolmea väriä. RGB-arvot esitetään yleensä kolmen väliltä 0-255 olevan numeron arvona (esimerkiksi **`50,78,255`**), kolmena prosenttiarvona (esimerkiksi **`75%,78%,99%`**) tai heksa-

desimaaliarvona (**A0B508**). Kaikissa kolmessa tavassa numerot ovat punaisen, vihreän ja sinisen määrä. Heksadesimaalina esitetyssä arvossa kaksi ensimmäistä kirjainta määrittelevät punaisen, kaksi seuraavaa vihreän ja kaksi viimeistä sinisen määrän.

Runko (WWW-sivun runko) WWW-sivun alue, jonne sijoitetaan varsinainen teksti, kuvat ym. sivun osat.

Selain (**Browser**). Ohjelma, jolla voi lukea WWW-sivuja. Selain osaa ottaa yhteyttä Internet-verkossa sijaitseviin WWW-palvelimiin ja hakea sivuja. Lisäksi selain osaa hakea sivuja myös muuntyyppisistä palvelimista (uutispalvelimet, **gopher**-palvelimet ja tiedostoarkistot) ja tietokoneen kovalevyltä.

Selitysteksti (Alt-teksti) Kuvakomentoon liitettävä teksti, jossa kerrotaan mikä teksti tulee näyttää niille lukijoille, joiden selain ei osaa näyttää kuvia.

Shockwave Formaatti multimediaohjelmille.

Solu (**Cell**) Taulukon osa. Taulukko koostuu riveistä ja sarakkeista, jotka taas koostuvat soluista.

Tapahtumankäsittelijä (**Event Handler**) JavaScript-ohjelmissa voidaan liittää tapahtumankäsittelijöitä erilaisiin sivulla oleviin objekteihin. Tällä tavalla voidaan esimerkiksi huomata, kun käyttäjä siirtää hiiren osoittimen linkin päälle.

Tarkennin (tiedostonnimen pääte, **extension**) Tiedoston nimen perässä oleva teksti, joka kertoo tiedoston tyypin. Esimerkiksi tiedoston **dokumentti.doc** tarkennin on **.doc**. HTML-tiedostojen tarkennin on **.html** tai **.htm**. Jälkimmäistä käytetään PC-koneissa, joissa tarkennin saa olla vain kolme merkkiä pitkä.

Taso (**Layer**) WWW-sivulla oleva alue, jonka sijainnin voi määritellä pikselin tarkkuudella mihin kohtaan ruudulla tahansa. Voit muokata tasojen avulla sivujesi ulkonäköä erittäin tarkasti.

Tekstilaatikko Lomakkeessa oleva kenttä, johon lukija voi kirjoittaa tekstiä.

Tiedostoarkisto Palvelin, joka pitää sisällään erilaisia tiedostoja. Julkiset tiedostoarkistot ovat kaikille avoimia: niihin on tallennettu esimerkiksi pelejä tai apuohjelmia, joita kuka tahansa voi käydä kopioimassa itselleen.

Tyylimäärittely (**Style Definition**) Määrittely, jolla kerrotaan miltä HTML-komennolla merkityn tekstin tulee näyttää kuvaruudulla. Voit esimerkiksi käskeä selainta näyttämään **<h1>** -komennoilla merkityt otsikot punaisina 5 senttimetriä korkeina otsikoina.

Tyylipohja (**Style Sheet**) Kokoelma tyylimäärittelyitä. Voit kytkeä tyylipohjan HTML-sivuun, jolloin sen kuvaruudulla näkyvä ulkoasu määräytyy tyylipohjassa annettujen ohjeiden perusteella. Tyylipohjaan tehdyt muutokset muuttavat myös kuvaruudulla näkyvän sivun ulkoasua.

URL (**Uniform Resource Locator**) Tiedon "osoite" Internetissä. URL voi osoittaa tiedostoon, WWW-sivuun, kuvaan, ääneen tai moneen muuhun mahdolliseen tietoon. URL kertoo tiedoston tarkan sijainnin. Sen avulla esimerkiksi selain osaa hakea halutun sivun.

Uutisryhmä (**Newsgroup**, keskusteluryhmä) Yksi Internetin palveluista. Internetin uutisryhmät ovat kuin 10 000-sivuinen sanomalehti, joka sisältää mielipideosastoja. Jokaisella sivulla on yksi tiettyä tarkkaa aihetta käsittelevä mielipideosasto, jonne kuka tahansa voi lähettää kirjoituksia. Eri sivujen aiheina voisivat olla vaikkapa "Hulluna sinuun" -TV-sarjan ystävien ajatustenvaihto, kas-

vissyönnistä keskustelu tai pohjoismainen kulttuuri. Uutisryhmät ovat näitä "mielipide-osastoja". Uutisryhmissä käydään keskustelua, joka koostuu tavallisten käyttäjien kirjoittamista artikkeleista sekä niihin lähetetyistä vastauksista. Kaikkialla maailmassa näkyviä uutisryhmiä on yli 10 000 kappaletta. Uutisryhmiä voi lukea joko erillisillä ohjelmilla ("uutistenlukuohjelmilla") tai WWW-selaimilla

WAV Ääniformaatti.

View Source Selaimen toiminto, jonka avulla sivun lukija voi katsoa sivulla käytettyjä HTML-komentoja. Toiminto löytyy yleensä selaimen valikoista.

WS-FTP Tiedostonsiirto-ohjelma.

WWW (*Web*) World Wide Web.

WWW-palvelin (Hypertekstipalvelin, **httpd**) Palvelin, joka tarjoaa tietoja selaimille.

Ylläpitäjä (**Webmaster**) Henkilö, joka pitää yllä palvelinta ja vastaa sen toiminnasta. Ylläpitäjä voi muuttaa palvelimen asetuksia ja muokata sen sisältämiä tiedostoja. Palvelimen ylläpitäjälle voi monissa tapauksissa lähettää sähköpostia osoitteella **webmaster@koneennimi**. Vaihda kohtaan *koneennimi* palvelimen Internet-osoite.

HTML-KOMENNOT

Alla on lueteltu yleisimmät HTML-komennot. Jokainen HTML-komento koostuu itse komennosta ja siihen mahdollisesti liitettävistä attribuuteista. HTML-komennolla voi olla myös lopetuskomento - komennon vaikutusalue ulottuu silloin aloitus- ja lopetuskomennon väliin, esimerkiksi:

```
<b>lihavoitua</b>
```

Komento voi sisältää yhtäaikaa monia attribuutteja - ne sijoitetaan komennon sisään välilyönneillä erotettuna. Attribuuteille voidaan antaa arvoja sijoittamalla attribuutin nimen jälkeen yhtäsuuruusmerkki ja attribuutin arvo. Arvo sijoitetaan usein lainausmerkkien sisään - tämä on välttämätöntä esimerkiksi silloin, kun attribuutin arvo sisältää välilyöntejä tai muita erikoismerkkejä. Esimerkki:

```
<img src="kuva.gif" width=400 height=200
alt="Yrityksen logo">
```

A

A-komennolla tehdään linkkejä ja linkkien kohteita. Komento ei tee yksinään mitään, vaan siihen lisätään tarvittaessa attribuutteja, jotka on lueteltu alla olevassa taulukossa. Linkin määrittely lopetetaan aina lopetuskomentoon ****.

name	Määrittelee kohteen kirjanmerkille. Esimerkki:
	
	`<a name="kohta">`***tekstiä***`</a>`
	
	Voit hypätä kohteeseen lisäämällä osoitteeseen tai tiedostonnimeen merkin # ja kirjanmerkin nimen:
	
	`<a href="tiedosto.html#kohta">` ***kuuma teksti***`</a>`

	Esimerkki lataa tiedoston **tiedosto.html** ja rullaa sitä selaimen ruudulla niin, että määritelty kohta on näkyvissä ruudulla.
href	Määrittelee kohteen linkkikomennolle. Kohde voi olla joko tiedostonnimi tai URL-osoite. Esimerkiksi: ** Linkki Yahoo:n sivuille**
target	Määrittelee kohteen linkille. Normaalisti linkkiä napsautettaessa seuraava sivu tuodaan samaan ikkunaan tai kehykseen. Kun linkkikomennossa annetaan **target**-attribuutti, sivu haetaan ikkunaan tai kehykseen, jolla on attribuutissa annettu nimi. Esimerkiksi: **<a href="navigointi.html" target="navigaatio">** Esimerkki lataa sivun **navigointi.html** kehykseen, jolle on annettu nimeksi **navigaatio**. Jos käytössä ei ole kehyksiä, selain avaa uuden selainikkunan ja antaa sille nimen **navigaatio**. Tämän jälkeen kaikki linkit, joiden kohteeksi on annettu nimi **navigaatio**, tulevat näkyviin tähän ikkunaan. Normaalisti nimien täytyy alkaa kirjainmerkeillä. Attribuutti voi saada kuitenkin joitain erikoisarvoja, joiden nimet alkavat alleviivausmerkillä: • **_blank**: Kohde avataan uuteen tyhjään ikkunaan. • **_self**: Kohde avataan samaan ikkunaan tai kehykseen, jossa itse linkkikomento on. • **_parent**: Kohde avataan hierarkiassa ylempänä olevaan kehykseen - siihen kehykseen, joka sisältää kehyksen, jossa linkkikomento on. Jos kehyksellä ei ole hierarkiassa ylempä-

	nä kehyksiä, attribuutti toimii samoin kuin **_self**. • **_top**: Kohde avataan koko ikkunan kokoisena. Jos linkki oli kehyksien sisällä, kehykset poistetaan näkyvistä. Jos kehykset eivät ole käytössä, tämä attribuutti toimii samoin kuin **_self**. Esimerkki: `<a href="http://www.yahoo.com/" target="_top">` Jos yllä oleva komento on kehyksen sisällä, **Yahoo** ei avaudu tämän kehyksen sisään vaan kehykset poistuvat käytöstä ja sivu avautuu koko ruudun kokoisena.

ADDRESS

Tekee kappaleen, jossa voidaan kertoa erilaisia osoitetietoja. Esimerkki:

```
<address>Matti.Meikäläinen@firma.fi</address>
```

APPLET

Sijoittaa sivulle Java-kielellä toteutetun ohjelman (**applet**). Aloitus- ja lopetuskomennon sijoitetaan **param**-komentoja, joilla välitetään parametrejä Java-ohjelmalle. Komentojen väliin voidaan sijoittaa myös tekstiä, joka tulostuu näytölle, kun selain ei osaa ajaa Java-ohjelmia.

Java-ohjelmille voi välittää parametrejä **applet**-komennon attribuuteilla:

codebase	Antaa ohjelman URL-osoitteen. Tämä on yleensä hakemisto, josta ohjelman koodi löytyy.
code	Tiedosto, joka sisältää ajettavan ohjelman. Normaalisti ohjelmaa etsitään samasta hakemistosta, jossa HTML-tiedosto sijaitsee, mutta sijaintia voidaan muuttaa **codebase**-attribuutilla.

alt	Teksti, joka näytetään silloin, kun selain ei osaa ajaa ohjelmaa. Tämän sijasta voidaan käyttää aloitus- ja lopetuskomennon välissä olevaa tekstiä.
width	Ohjelman ikkunan leveys ja korkeus pikseleinä.
height	
align	Ohjelman ikkunan sijoittelu näytöllä - tätä käytetään samalla tavalla kuin kuvakomentojen **align**-attribuuttia.
hspace	Ohjelman ja ympäröivän tekstin väliin jätettävä tyhjä tila pikseleinä. **Hspace** määrittelee vaakasuoran tilan ja **vpace** pystysuoran.
vspace	
name	Ohjelmalle annettava nimi.

AREA

Määrittelee interaktiivisen kuvan osan. Area-komennot sijoitetaan **map**-komentojen sisään. Yksi komento määrittelee yhden kuvion interaktiivisessa kuvassa. Komennon attribuuteilla voidaan määritellä alueen muoto, koko ja sen sisällä olevan linkin osoite. **Area**-komennolla ei ole lopetuskomentoa.

shape	Interaktiivisen alueen muoto. Tämä voi olla joko **rect**, **circle** tai **poly**. Esimerkki:
coords	Alueen koon määrittelevät koordinaatit.
href	Kertoo tiedostonnimen tai osoitteen, johon selain siirtyy aluetta napsautettaessa.
nohref	Kertoo, että tämän alueen napsauttaminen ei tee mitään.
alt	Vapaamuotoinen teksti, joka kuvailee aluetta. Selain saattaa näyttää tämän tekstin käyttäjälle hänen liikutellessaan hiiren osoitinta alueen päällä.

Esimerkki:

```
<area shape="rect" coords="13,13,95,53"
href="http://www.cnet.com/">
```

B

Tekee lihavoitua tekstiä. Esimerkki:

```
Tavallista tekstiä, <b>lihavoitua</b> ja taas
tavallista.
```

BASE

Määrittelee osoitteen, joka lisätään sivulla olevien relatiivisten osoitteiden alkuun. Relatiiviset osoitteet ovat tiedostojen nimiä joko hakemistojen kanssa tai ilman niitä ilmaistuna. Komento sijoitetaan sivun **head**-komentojen väliin.

Base-komennolle annetaan attribuutti, **href**, jossa annetaan haluttu URL-osoite. Sillä ei ole lopetuskomentoa.

Normaalisti seuraavat komennot lataisivat HTML-tiedoston kanssa samassa hakemistossa olevan kuvan ja hyppäisivät samassa hakemistossa olevaan tiedostoon:

```
<img src="kuva.gif">
<a href="tiedosto.html">kuuma teksti</a>
```

Jos lisäät sivun alkuun **base**-komennon, selain lisää tämän osoitteen aina annettujen tiedostojen nimien eteen:

```
<html>
<head>
<base href="http://www.yahoo.com/">
</head>
<body>
<img src="kuva.gif">
<a href="tiedosto.html">kuuma teksti</a>
</body>
```

Selain yrittää hakea kuvaa osoitteesta

```
http://www.yahoo.com/kuva.gif
```

ja hypätä linkkiä napsautettaessa osoitteeseen

```
http://www.yahoo.com/tiedosto.html
```

riippumatta siitä missä HTML-tiedosto sijaitsee.

Base-komento voi saada myös attribuutin **target**. Tämä attribuutti toimii samoin kuin linkkikomennon (**A**) attribuutti **target**, mutta vaikuttaa kaikkiin sivulla oleviin linkkeihin yhtäaikaa (paitsi niihin, joissa on oma **target**-attribuutti).

BASEFONT

Määrittelee tavallisen tekstin koon. Komento saa yhden attribuutin, **size**, joka kertoo, millä koolla sivulla näkyvä teksti tulostetaan. Koko on numero väliltä 1-7 (1 on pientä tekstiä, 7 suurinta). Normaalikoko on 3. **Basefont** ei sisällä lopetuskomentoa.

```
<basefont size=5>
<p>Tämä on suurta tekstiä
```

BGSOUND

Määrittelee sivun taustalla soivan äänen. Komennolle annetaan attribuutit **src** (joka kertoo tiedostonnimen tai osoitteen, josta ääni löytyy) ja **loop** (joka kertoo, montako kertaa äänitiedosto toistetaan).

BIG

Tulostaa tekstiä normaalia suuremmalla koolla. Suurennettava teksti sijoitetaan aloitus- ja lopetuskomennon väliin. Esimerkki:

```
<p>Normaalia tekstiä, <big>suurta tekstiä</big>.
```

BLINK

Aloitus- ja lopetuskomennon välissä oleva teksti määritellään vilkkuvaksi.

```
<blink>Erikoistarjous</blink>
```

BLOCKQUOTE

Tekee kappaleesta lainauksen. Tällä voit esittää muista lähteistä lainattua tekstiä. Lainattu teksti sijoitetaan aloitus- ja lopetuskomennon väliin. Välissä voi olla myös muita HTML-komentoja.

```
<blockquote>
Voi aikoja, voi tapoja!
</blockquote>
```

BODY

Määrittelee WWW-sivun selaimessa näkyvän osan. Ennen tätä komentoa voidaan määritellä sivun sisäisiä tietoja **head**-komentojen avulla. Selaimessa näkyvä asia sijoitetaan aloitus- ja lopetuskomennon väliin.

```
<html>
<head>
<title>Sivun nimi</title>
</head>
<body>
HTML-komentoja
</body>
</html>
```

Body-komennon attribuuteilla voidaan vaikuttaa selaimessa näkyvän sivun väreihin:

bgcolor	Määrittelee sivun taustavärin. Esimerkiksi: `<body background="White">`

text	Määrittelee sivun tekstin värin.
link	Määrittelee sivulla olevien linkkien värin. Tämä väri vaikuttaa linkkeihin, joissa lukija ei ole ennen käynyt.
vlink	Tämä attribuutti määrittelee värin sellaisille linkeille, joissa käyttäjä on käynyt ennenkin.
alink	Kun linkkiä napsautetaan hiirellä, sen väri vaihtuu hetkeksi. Tämä attribuutti määrittelee silloin näkyvän värin.
background	Määrittelee sivun taustakuvan. Tämä voi olla **gif**- tai **jpeg**-kuvaan osoittava tiedostonnimi tai URL-osoite.

BR

Tekee rivinvaihdon. Komennolle voidaan antaa attribuutti **clear**, jolloin rivinvaihto ulottuu sivun reunalle sijoiteltujen kuvien alle.

Attribuutti **clear=left** ohittaa sivun vasempaan reunaan sijoitetut kuvat, **clear=right** oikeaan reunaan sijoitetut kuvat ja **clear=both** molempiin reunoihin sijoitetut kuvat. **Br**-komennolla ei ole lopetuskomentoa.

Esimerkki 1:

```
<p>Maija Meikäläinen<br>
Selaintie 18
90210 Operaattorila
```

Esimerkki 2:

```
<p><img src="kuva.gif" align=left> Tämä teksti tulee
kuvan viereen.
<br clear=left>
Tämä teksti tulee kuvan alle.
```

CAPTION

Määrittelee taulukolle annettavan otsikkotekstin. Attribuutilla **align** kerrotaan, tuleeko otsikkoteksti taulukon ylä- vai alapuolelle. Attribuutti voi saada arvot **top** tai **bottom**. Itse otsikkoteksti sijoitetaan aloitus- ja lopetuskomennon väliin.

Esimerkki:

```
<caption align=bottom>Taulukko 8: Läänien
väkiluku</caption>
```

CENTER

Keskittää tekstin ja kuvat. Sijoita keskitettävä materiaali aloitus- ja lopetuskomennon väliin:

```
<center>
<h1>Keskitetty otsikko</h1>
<p>Keskitettyä tekstiä ja kuva. <img src="kuva.gif">
</center>
```

CITE

Määrittelee tekstin sitaatiksi. Sijoita määriteltävä teksti aloitus- ja lopetuskomennon väliin:

```
Lähde: <cite>Programming Perl</cite>.
```

CODE

Määrittelee tekstin näyttämään ohjelmakoodilta. Sijoita määriteltävä teksti aloitus- ja lopetuskomennon väliin:

```
Voit tulostaa muuttujan komennolla
<code>document.write(muuttuja);</code>.
```

DD

Tätä käytetään määritelmälistojen yhteydessä. Komennolla annetaan termin määritelmä. Komennolle voi antaa lopetuskomennon, mutta se ei ole välttämätön:

```
<dl>
<dt>Kaktus</dt>
<dd>Piikikäs kasvi</dd>
</dl>
```

DFN

Määrittelee tekstin roolin jonkin asian määrittelyksi. Sijoita teksti aloitus- ja lopetuskomentojen väliin:

```
<dfn>JavaScript</dfn> on ohjelmointikieli, jolla voit
lisätä WWW-sivujesi interaktiivisuutta.
```

DIR

Toimii kuten ranskalaisten viivojen lista. Aloitus- ja lopetuskomentojen välissä annetaan lista asioita, jotka tulee esittää listausmuodossa:

```
<dir>
<li>Aatami
<li>Beetami
</dir>
```

DIV

Määrittelee osan sivusta. Aloitus- ja lopetuskomentojen välissä oleva osa voidaan sijoitella oikealle tai vasemmalle tasatuksi tai keskitetyksi komennon attribuutilla **`align`**. Attribuutti voi saada arvot **`left`**, **`center`** tai **`right`**.

Esimerkki:

```
<div align=right>
<h1>Oikeaan reunaan tasattu otsikko</h1>
<p>Oikeaan reunaan tasattua tekstiä ja kuva
<img src="kuva.gif">
</div>
```

DL

Aloittaa ja lopettaa määritelmälistan. Itse lista sijoitetaan aloitus- ja lopetuskomennon väliin:

```
<dl>
<dt>Kaktus</dt>
<dd>Piikikäs kasvi</dd>
</dl>
```

Komennolle voidaan antaa attribuutti **compact**, jolloin selain yrittää tulostaa listan tiiviimmässä muodossa.

DT

Kertoo määritelmälistassa olevan termin. Komennolle voidaan antaa lopetuskomento, mutta se ei ole välttämätön.

```
<dl>
<dt>Kaktus</dt>
<dd>Piikikäs kasvi</dd>
</dl>
```

EM

Korostaa tekstiä. Sijoita korostettava teksti aloitus- ja lopetuskomennon väliin.

```
<p>On <em>tärkeää</em> muistaa eräitä tiettyjä
asioita.
```

EMBED

Tällä komennolle sijoitetaan WWW-sivulle tiedostoja, joille ei ole normaaleja lisäyskomentoja: ääniä, videoleikkeitä tai multimediatiedostoja. Selain näyttää tiedostot käyttäen apuna ulkopuolista apuohjelmaa, **plug-iniä**.

Komento voi saada seuraavia attribuutteja:

align	Toimii kuten kuvien yhteydessä: Tällä voidaan määritellä **plug-in**-alueen sijainti suhteessa tekstiin. Mahdollisia arvoja ovat **left**, **right**, **top** ja **bottom**.
border	Määrittelee **plug-in**-alueen reunan paksuuden pikseleinä.
height	Määrittelee **plug-in**-alueen koon pikseleinä. Tiedoston sisältö ja mahdolliset **plug-in**-ohjelman kontrollipainikkeet tulevat näkyviin tämän kokoisina.
width	
hspace	Määrittelee **plug-inin** ja muun tekstin väliin tulevan reunan paksuuden pikseleinä.
vspace	
hidden	Piilottaa **plug-inin** ja sen kontrollit. Tästä on hyötyä esimerkiksi ääniä soittavan **plug-in**in kanssa. Voi saada arvon **true** tai **false**.
frameborder	Määrittelee kehyksen reunan paksuuden.
pluginspace	Määrittelee osoitteen, josta **plug-in**-ohjelma voidaan hakea, jos selaimessa ei ole sellaista valmiina.
name	Antaa nimen **plug-in**-alueelle.
src	Määrittelee tiedoston tai URL-osoitteen, josta näytettävä (tai soitettava) tiedosto haetaan.
type	Määrittelee näytettävän (tai soitettavan) tiedoston mediatyypin siltä varalta, että selain ei saa sitä selville tiedoston nimestä.

Komento voi saada lisäksi muitakin attribuutteja riippuen käytetystä **plug-in**-sovelluksesta. Lisätietoja näistä saat **plug-inin** ohjeista.

FONT

Määrittelee kirjasintyypin koon, värin ja ulkoasun. Määriteltävä teksti sijoitetaan aloitus- ja lopetuskomennon väliin.

Font-komennolle voidaan antaa attribuutteja, jotka vaikuttavat tekstin ulkoasuun:

size	Kirjasintyypin suhteellinen koko. Koko annetaan joko numeroarvona 1-7 (1 on pienin, 7 suurin, 3 on normaalin tekstin koko) tai suhteellisena arvona. Esimerkki: **Pienellä painettua Yhtä pykälää isompaa kuin normaaliteksti**
color	Tekstin väri. Voidaan antaa joko värin nimenä tai **RGB**-arvona. Esimerkkejä: **Punaista tekstiä Sinistä tekstiä**
face	Tekstityyppi. Tässä voidaan antaa lista fonttien nimiä - selain käyttää pyydettyä fonttia, jos sellainen löytyy lukijan tietokoneesta. Jos nimiä on annettu useampia, selain yrittää käyttää niitä alusta loppuun päin - jos ensimmäistä ei löydy, se yrittää toista jne. **Helvetica-tekstiä (toivottavasti)**

FORM

Aloittaa ja lopettaa lomakkeen. Kaikki lomakkeen komennot (painikkeet, tekstikentät ja valikot) täytyy sijoittaa aloitus- ja lopetuskomentojen väliin. **Form**-komento voi saada seuraavia attribuutteja:

action	Määrittelee, minne lomakkeelta syötetyt tiedot lähetetään, kun käyttäjä painaa **submit**-painiketta. Tässä voi olla CGI-ohjelman osoite tai **mailto**-osoite: **<form action="http://palvelin/cgi-bin/ohjelma">** **<form action="mailto:pekka@palvelin.fi">**
method	Määrittelee, missä muodossa lomakkeelta syötetyt tiedot lähetetään. Vaihtoehtoja ovat **GET** ja **POST**. **GET**-muodossa lähetetyt tiedot sijoitetaan action-kohdassa määritellyn osoitteen perään, **POST**-muodossa lähetetyt tiedot välitetään ohjelmalle erillistä siirtotietä pitkin.
enctype	Määrittelee miten lähetettävät tiedot koodataan. Normaalisti tätä ei aseteta, vaan käytetään oletusarvoa **application/x-www-form-urlencoded**.

FRAME

Frame-komento määrittelee kehyksen sisällön. Komento sijoitetaan **frameset**-komentojen väliin. Jokaista kehystä kohti annetaan yksi **frame**-komento. Esimerkiksi:

```
<html>
<head><title>Frame-esimerkki</title></head>
<frameset rows="33%,33%,*">
 <frame src="http://www.uta.fi/">
 <frame src="http://www.yahoo.com/">
 <frame src="http://www.news.com/">
</frameset>
<noframes>
<p>Valitettavasti ohjelmasi ei tue kehyksiä!
</noframes>
</html>
```

Komennolla ei ole lopetuskomentoa. Attribuutilla **src** annetaan tiedostonnimi tai osoite, josta kehyksen sisältö haetaan. Muita attribuutteja ovat:

name	Antaa kehykselle nimen. Tällä tavalla muissa ikkunoissa olevat linkit voivat vaihtaa tämän kehyksen sisältöä.
noresize	Kertoo selaimelle, että käyttäjä ei voi muuttaa tämän kehyksen kokoa hiirellä. Normaalisti käyttäjä voi siirtää kehyksien reunoja hiiren osoittimella.
scrolling	Jos kehyksen sisältö ei mahdu kerralla näkyviin, selain tuo oletusarvoisesti vierityspalkit kehyksen reunaan. Tällä attribuutilla voidaan määritellä vierityspalkkien näkymistä. Arvo **yes** pakottaa vierityspalkit näkymään aina (vaikka niitä ei tarvittaisikaan), **no** kieltää vierityspalkkeja näkymästä silloinkaan, kun niitä tarvittaisiin. Arvo **auto** on oletusarvo: vierityspalkit tuodaan näkyviin tarvittaessa.
frameborder	Kertoo selaimelle, piirretäänkö kehykselle reuna. Arvo 1 (oletusarvo) kertoo, että reuna piirretään, arvo 0 kieltää reunojen piirtämisen.
marginwidth	Määrittelee kehyksen pystyreunan ja sisällön väliin tulevan marginaalin koon pikseleinä.
marginheight	Määrittelee marginaalin kehyksen ylä- ja alareunojen ja sisällön välissä.

FRAMESET

Frameset-komennolla määritellään selaimen ikkunassa näkyvät kehykset. Aloitus- ja lopetuskomento sijoitetaan sivulle **body**-komentojen paikalle. Komentojen väliin sijoitetaan **frame**-komentoja, jotka määrittelevät itse kehysten sisällön tai uusia **frameset**-komentoja, joilla yksittäinen kehys voidaan jakaa edelleen osiksi.

Komento voi saada seuraavia attribuutteja:

rows	Jakaa ruudun osiin pystysuunnassa. Attribuutti voi saada arvot joko pikselimäärinä, prosenttiarvona selaimen ikkunasta tai suhteellisena arvona. Esimerkkejä:

	<frameset rows="50%,50%"> Ruutu jaetaan kahteen yhtä suureen osaan pystysuunnassa. **<frameset rows="200,*">** Ruutu jaetaan kahteen osaan: ensimmäinen on korkeudeltaan 200 pikseliä ja toinen osa saa kaiken lopputilan. **<frameset rows="4*,1*">** Ruutu jaetaan kahteen osaan: ensimmäinen saa 4/5 ikkunan korkeudesta ja toinen 1/5.
cols	Kuten **rows**, mutta jakaa ruudun rinnakkaisiin osiin.

H1-H6 (OTSIKOT)

Määrittelee otsikon. Aloitus- ja lopetuskomennon väliin sijoitetaan otsikoksi tarkoitettu teksti. Komennolla **H1** tehdään suurikokoisin otsikko ja komennolla **H6** pienikokoisin. Lopetuskomennon täytyy sisältää sama numero kuin aloituskomennonkin.

```
<h2>Väliotsikko</h2>
```

Otsikkokomentoon voidaan lisätä attribuutti **align**, jolla voi sijoittaa otsikon tasatuksi vasempaan reunaan (**align=left**), keskellä (**align=center**) tai oikeaan reunaan (**align=right**).

HEAD

Määrittelee sivun sisäisiä tietoja. Sijoitetaan tiedoston alkuun ennen **body**-komentoa. Aloitus- ja lopetuskomennon väliin voidaan sijoittaa komentoja **title**, **isindex**, **base**, **meta**, **link**, **style** ja **script**.

```
<html>
<head>
<title>Kotisivu</title>
</head>
<body>
<h1>Tervetuloa kotisivulleni!</h1>
</body>
</html>
```

HR

Tekee kuvaruudulle viivan. Tällä tavalla voit erotella sivun osia toisistaan. Tällä komennolla ei ole lopetuskomentoa.

Yksittäinen **hr**-komento tekee ikkunan laidasta laitaan ulottuvan viivan. Voit muokata viivan ulkonäköä attribuuteilla:

size	Viivan korkeus pikseleinä: `<hr size=8>`

width	Viivan leveys pikseleinä tai prosenttiarvona selaimen ikkunan leveydestä:
	
	`<hr width="50%">`
	`<hr width=100>`

align	Jos viiva ei ole koko ruudun levyinen, tällä voidaan määritellä viivan vaakasuora sijainti: **left**, **center** tai **right**. `<hr width="25%" align=right>`
noshade	Tekee viivasta koristelemattoman - pelkän viivan. Normaalisti selain näyttää viivan varjostettuna.

HTML

Komento, joka sijoitetaan tiedoston alkuun ja loppuun:

```
<html>
<head>
<title>Kotisivu</title>
</head>
<body>
<h1>Tervetuloa kotisivulleni!</h1>
</body>
</html>
```

Komennolle voidaan antaa attribuutti **version**, joka kertoo käytettävän HTML-version.

I

Määrittelee kursiivitekstin. Sijoita kursivoitava teksti aloitus- ja lopetuskomentojen väliin:

```
<p>Normaalitekstiä, <i>kursiivia</i>.
```

IFRAME

Iframe-komennolla voi sijoittaa WWW-sivun sisään kehyksiä. Komento sisältää aloitus- ja lopetuskomennon. Niiden väliin voidaan sijoittaa tekstiä, joka tulee näkyviin selaimissa, jotka eivät osaa **iframe**-komentoa.

Komento saa seuraavat attribuutit:

width	Kehyksen leveys ja korkeus
height	
src	Kertoo tiedoston tai URL-osoitteen, josta kehyksen sisältö haetaan.
scrolling	Määrittelevät kehyksen ulkoasun samalla tavalla kuin **frame**-komentoa käytettäessä.
marginwidth	
marginheight	

frameborder	
align	Määrittelee kehyksen sijainnin suhteessa tekstiin, kuten kuvakomennoissa.

ILAYER

Toimii kuten **layer**-komento. Tuo tason näkyviin ruudulla olevan tekstin väliin eikä päälle kuten **layer**-komento. Komennon attribuutit ovat samat kuin **layer**-komennossa.

IMG

Sijoittaa sivulle kuvan. Kuvatiedoston sijainti määritellään attribuutilla **src**, jolle voi antaa arvoksi tiedostonnimen tai URL-osoitteen:

```
<img src="kuva.gif">
```

Img-komennolla ei ole lopetuskomentoa. Muita kuvakomentoon liitettäviä attribuutteja ovat:

alt	Määrittelee selitystekstin kuvalle. Selaimet näyttävät tämän tekstin kuvan paikalla, jos ne eivät voi syystä tai toisesta näyttää kuvaa. `<img src="kuva.gif" alt="Yrityksen logo">`
align	Määrittelee kuvan sijainnin suhteessa muuhun tekstiin. Mahdollisia arvoja ovat: • **top**: kuvan yläreuna on tekstin kanssa samalla korkeudella. • **middle**: kuvan keskikohta on tekstin kanssa samalla korkeudella. • **bottom**: kuvan alareuna on tekstin kanssa samalla korkeudella.

	• **left**: kuva sijoitetaan ikkunan vasempaan reunaan ja kuvan jälkeen tuleva teksti juoksee sen rinnalla. • **right**: kuva sijoitetaan ikkunan oikeaan reunaan ja kuvan jälkeen tuleva teksti juoksee sen rinnalla.
height	Kertoo selaimelle kuvan korkeuden pikseleinä. Selain voi käyttää tätä tietoa hyväkseen varatessaan kuvalle tilaa sivua ladattaessa. Korkeus voi olla myös jokin muu kuin kuvan oikea korkeus: silloin kuvaa venytetään.
width	Kuten **height**, mutta määrittelee kuvan leveyden.
border	Kun kuva on linkin kuumana tekstinä, sen ympärille piirretään linkin värinen reuna. Tämä attribuutti määrittelee pikseleinä reunan paksuuden.
hspace	Kertoo pikseleinä kuvan ympärille vaakasuunnassa jätettävän tyhjän tilan koon.
vspace	Kuten **hspace**, mutta pystysuunnassa.
usemap	Kertoo, että kuva on interaktiivinen ja määrittelee käytettävän kuvakartan.
ismap	Määrittelee vanhemmantyylisen interaktiivisen kuvan. Kuvan ympärille täytyy tehdä tätä käytettäessä linkki **imagemap**-ohjelmaan.

INPUT

Määrittelee lomakkeen elementin. Tällä komennolla ei ole lopetuskomentoa. Input-komentoja voidaan käyttää vain lomakkeiden sisällä.

Komennon attribuutit määrittelevät tarkemmin elementin tyypin:

type	Kertoo elementin tyypin. Attribuutti voi saada seuraavat arvot:

	• **text**: tekstikenttä. • **password**: tekstikenttä, johon syötetty teksti ei tule näkyviin. • **checkbox**: rasti ruutuun -painike. • **radio**: radiopainike. • **submit**: painike, joka lähettää lomakkeen tiedot käsiteltäviksi. • **reset**: painike, joka nollaa lomakkeeseen syötetyt tiedot alkuarvoihinsa. • **file**: kenttä, johon käyttäjä voi syöttää tiedostonnimen. Annettu tiedosto lähetetään lomakkeen muiden kenttien mukana. Ei toimi kaikissa selaimissa! • **hidden**: kenttä, joka ei tule näkyviin lukijalle. • **image**: kenttä, joka sisältää kuvan. Käyttäjä voi napsauttaa kuvaa tietystä kohdasta, jolloin ko. kohdan tiedot lähetään lomaketta käsittelevälle ohjelmalle kahdessa muuttujassa. Muuttujien nimet tulevat **name**-attribuutista: **name.x** ja **name.y**. • **button**: tyhjä nappi. Tähän voidaan liittää toiminnallisuutta JavaScript-komennoilla.
name	Elementin nimi. Tämä välitetään eteenpäin lomakkeen tietoja käsittelevälle ohjelmalle.
value	Elementin oletusarvoinen sisältö. **Submit**- ja **reset**-tyyppisissä painikkeissa määrittelee näytöllä olevan painikkeen tekstin. **Radio**- ja **checkbox**-painikkeissa määrittelee käsittelevälle ohjelmalle siirtyvän arvon. Tekstikentissä määrittelee tekstin, joka näkyy valmiiksi syötettynä tekstikentän sisään.

checked	**Radio**- ja **checkbox**-painikkeissa määrittelee painikkeen valituksi: **<input type=radio name="sukupuoli" value="nainen" checked>**
size	Tekstikentän ruudulla näkyvä koko (kirjaimina).
maxlength	Maksimimäärä kirjaimia, joka voidaan syöttää tekstikenttään.
src	**Image**- ja painiketyyppisissä elementeissä määrittelee kuvatiedoston, josta painikkeen ulkoasu haetaan.
align	Määrittelee sijoittelun sellaisissa painikkeissa, joiden ulkonäköä on vaihdettu **src**-attribuutilla. Voi saada samat arvot kuin **img**-komennon vastaava attribuutti.

ISINDEX

Tekee sivun alkuun tekstikentän, johon käyttäjä voi syöttää hakusanoja. Sijoitetaan **head**-komentojen väliin. Tällä ei ole lopetuskomentoa.

Komento itsessään ei tee sivusta vielä haettavaa: sivuun täytyy liittää ohjelma, joka hoitelee varsinaisen hakemisen. Kun käyttäjä kirjoittaa **isindex**-kenttään hakusanan ja painaa return-näppäintä, selain kutsuu samaa sivua uudelleen mutta lisää pyytämänsä URL-osoitteen perään kysymysmerkin ja annetun hakusanan.

Attribuutilla **prompt** voidaan määritellä käyttäjälle näkyvä kysymysteksti.

KBD

Määrittelee aloitus- ja lopetuskomennon välissä olevan tekstin esimerkiksi näppäilyistä:

```
<p>Näppäile <kbd>dir</kbd>, niin saat
tiedostolistauksen.
```

LAYER

Layer-komennolla määritellään selaimessa näkyvä taso. Aloitus- ja lopetuskomentojen väliin sijoitetaan tason sisältö. Komento voi saada seuraavia attribuutteja:

width **height**	Määrittelevät tason leveyden ja korkeuden. Arvon voi antaa pikseleinä tai prosenttiarvona.
top **left**	Määrittelevät tason paikan ruudulla. Jos taso on toisen tason sisällä, paikka mitataan edellisen tason vasemmasta yläkulmasta, muussa tapauksessa sivun vasemmasta yläkulmasta.
bgcolor	Määrittelee tason taustavärin.
background	Määrittelee tason taustakuvan.
id	Antaa tasolle nimen.
pagex **pagey**	Määrittelevät tason sijainnin suhteessa sivun vasempaan yläkulmaan.
src	Määrittelee tason sisällön. Attribuutti voi saada arvokseen joko tiedoston nimen tai URL-osoitteen.
clip	Määrittelee tason leikkauksen. Tällä tavalla tasosta voidaan määritellä näkymään vain jokin tietty osa. Attribuutille annetaan kaksi tai neljä parametriä, jotka kertovat näkyviin jäävän kohdan vasemman yläkulman ja oikean alakulman (4 parametriä) tai pelkän oikean alakulman (2 parametriä - vasen yläkulma on tässä tapauksessa tason vasemmassa yläkulmassa)
above **below**	Määrittelee tasojen järjestyksen silloin, kun ne ovat päällekkäin. **Above** ja **below** saavat arvokseen tason nimen, **z-order** saa numeron (tasot kasataan näytölle numerojärjestyksessä - suurin numero tulee päällimmäiseksi)

z-order	
visibility	Määrittelee tason näkymisen. Arvoja ovat **show** (näkyvissä), **hide** (ei näkyvissä), **inherit** (sama kuin edellisellä tasolla, jos kyseessä on tason sisällä oleva taso).

LI

Määrittelee listan kohdan määritelmä-, numero- ja ranskalainen viiva -listoissa. Esimerkiksi:

```
<ul>
<li>Omenoita
<li>Appelsiineja
<li>Banaaneja
</ul>
```

Komento voi saada attribuutteja:

type	Määrittelee ruudulla näkyvien listan kohtien ulkoasun. Ranskalainen viiva -listoissa attribuutti voi saada arvot **disc**, **square** tai **circle**, jotka kertovat listan kohdan eteen tulevan symbolin ulkonäön. Numerolistoissa arvoja voivat olla **1** (normaali numerolista); **A** (kirjainlista suurilla kirjaimilla: A, B, C, ...); **I** (roomalaiset numerot: I, III, III, ...) tai **i** (roomalaiset numerot pienillä kirjaimilla: i, iii, iii, ...).
value	Määrittelee kohdan numeron (numerolistoissa).

LINK

Määrittelee sivun suhteen toisiin sivuihin. Tämä komento sijoitetaan sivujen **head**-osaan eikä sillä ole lopetuskomentoa.

Nykyisissä selaimissa tällä komennolla on varsin vähän käyttöä. Mahdollisia attribuutteja ovat:

rel	Määrittelee "eteenpäin" osoittavan relaation. Tätä käytetään esimerkiksi liitettäessä tyylimäärittelyitä sivulle: **<link rel=stylesheet type="text/css" href="tyyli.css">**
rev	Määrittelee "taaksepäin" osoittavan relaation. Esimerkiksi: **<link rev=made href="mailto:tekijä@osoite.fi">**
title	Kertoo relaation kohteen nimen.
type	Kertoo linkin kohteen tyypin.
href	Kertoo linkin kohteen osoitteen tai tiedostonnimen.

MAP

Määrittelee kuvakartan. Aloitus- ja lopetuskomennon väliin sijoitetaan **area**-komentoja, jotka kertovat kuvan kuumat kohdat. **Map**-komennolle annetaan attribuutti **name**, jolla sille annetaan nimi (tämän nimen avulla kuvakarttaan viitataan **img**-komennosta).

```
<map name="kuviot">
<area shape="circle" coords="95,48,33"
href="kotisivu.htm">
<area shape="rect" coords="13,13,95,53"
href="http://www.cnet.com/">
<area shape="poly" coords="129,81, 130,62, 143,45,
144,26, 163,1 184,1, 195,24, 144,83"
href="http://www.yahoo.com/">
</map>
```

MENU

Tekee ranskalaisia viivoja muistuttavan listan. Aloitus- ja lopetuskomennon väliin sijoitetaan **li**-komennoilla listan kohtia. Attribuutilla **compact** voidaan pyytää selainta pakkaamaan lista pienempään tilaan.

```
<menu compact>
<li>Hip
<li>Hop
<li>Hei
</menu>
```

META

Meta-komennolla annetaan selaimelle tai muille ohjelmille informaatiota sivusta. Komento sijoitetaan **head**-komentojen väliin. Sillä ei ole lopetuskomentoa.

Komento voi saada seuraavia attribuutteja:

name	Määrittelee tiedon tyypin. Esimerkiksi: `<meta name=KEYWORDS content="kalastus, vieheet">` Yllä oleva esimerkki tarjoaa hakukoneille tietoa sivun avainsanoista.
content	Tiedon sisältö.
http-equiv	Vastaa **name**-attribuuttia. Esimerkki: `<META HTTP-EQUIV="Refresh" CONTENT="20; URL=index2.html">` Saa sivun siirtymään osoitteeseen **index2.html** kahdenkymmenen sekunnin kuluttua.

NOBR

Aloitus- ja lopetuskomennon väliin sijoitetaan tekstiä, joka on tarkoitus pitää aina samalla rivillä. Selain pitää määritellyn tekstin yhdellä rivillä eikä katkaise sitä - vaikka lopputuloksena olisi sekin, että teksti ei mahdu kerralla näytölle.

```
Puhelinnumero: <nobr>(03) 123 4567</nobr>
```

Tekstin sisään voidaan määritellä kohtia, joista teksti voidaan katkaista komennolla **wbr**.

```
<nobr>Creating Killer Websites<wbr> by David
Siegel</nobr>
```

Jos rivinvaihtoa ei tarvita tuossa kohdassa, selain jättää komennon huomiotta.

NOEMBED

Määrittelee aloitus- ja lopetuskomennon välissä tekstin, joka näytetään, kun selain ei osaa käyttää **embed**-komentoa.

```
<noembed>Valitettavasti tällä sivulla olevat
videoleikkeet eivät toimi sinulla koska selaimesi ei
ymmärrä tarvittavia komentoja</noembed>
```

NOFRAMES

Noframes-komennolla määritellään teksti, joka tulee näkyviin kehyksien sijasta sellaisissa selaimissa, jotka eivät osaa näyttää kehyksiä. Teksti sijoitetaan aloitus- ja lopetuskomentojen väliin. Komennot voidaan sijoittaa joko **frameset**-komentojen sisään tai niiden jälkeen. Esimerkki:

```
<html>
<head><title>Frame-esimerkki</title></head>
<frameset rows="33%,33%,*">
 <frame src="http://www.uta.fi/">
 <frame src="http://www.yahoo.com/">
 <frame src="http://www.news.com/">
</frameset>
<noframes>
<p>Valitettavasti ohjelmasi ei tue kehyksiä!
</noframes>
</html>
```

NOLAYER

Aloitus- ja lopetuskomentojen välissä oleva teksti tulee näkyviin selaimessa, jos se ei osaa näyttää tasoja.

```
<nolayer>Valitettavasti selaimesi ei osaa näyttää
tasoja!</nolayer>
```

NOSCRIPT

Aloitus- ja lopetuskomennon väliin sijoitetaan tekstiä, joka tulee näkyviin silloin, kun käyttäjän selain ei pysty ajamaan JavaScript-ohjelmia. Komento sijoitetaan **head**-komentojen väliin.

OL

Aloittaa numeroidun listan. Aloitus- ja lopetuskomentojen väliin sijoitetaan **li**-komennoilla listan kohtia:

```
<ol>
<li>Nouse ylös
<li>Pese hampaat
<li>Keitä kahvia
<li>Juo kahvi
</ol>
```

Komento voi saada attribuutteja, jotka vaikuttavat listan ulkoasuun:

type	Listan kohtien ulkoasu. Voi saada seuraavia arvoja: • **1**: Normaali numeroitu lista (1,2,3,...) • **A** tai **a**: Aakkoslista joko isoilla tai pienillä kirjaimilla (A, B, C, ... tai a, b, c, ...). • **I** tai **i**: Roomalaisilla numeroilla numeroitu lista joko isoilla tai pienillä kirjaimilla: (I, III, III, … tai i, ii, iii, …)

start	Listan aloitusnumero
compact	Pyytää selainta näyttämään listan pienemmässä tilassa, jos mahdollista.

OPTION

Määrittelee lomakkeen valikon kohdat. Sijaitsee lomakkeen sisällä **select**-komentojen välissä. Lopetuskomento ei ole välttämätön. Esimerkki:

```
<select name="valikko">
<option>Suolaa
<option>Sokeria
<option>Chilipippuria
</select>
```

Yhdelle valikon kohdalle voi antaa attribuutin **selected**, joka määrittelee tämän kohdan oletusarvoisesti valituksi, kun valikko tuodaan näkyviin.

Normaalisti lomaketta lähettäessä valikko palauttaa sen tekstin, joka on valikossa **option**-komennon jälkeen, esimerkiksi:

```
valikko=Sokeria
```

Palautettavaa arvoa voi muuttaa **value**-attribuutilla:

```
<option value="fi">Suomi
```

P

Määrittelee leipätekstikappaleen alun. Kappaleen loppuun voidaan sijoittaa lopetuskomento **</p>**, mutta se ei ole välttämätön.

```
<p>Tervetuloa kotisivulleni!
```

Komennolle voidaan antaa attribuutti **align**, joka määrittelee kappaleessa olevan tekstin sijoittelun. Attribuutti voi saada arvot **left**, **center** tai **right**.

```
<p align=right>(C)1997 Veikko Webmaster
```

PARAM

Määrittelee Java-ohjelmalle välitettävät parametrit. Komento sijoitetaan **applet**-komennon sisään. Komennolle annetaan kaksi attribuuttia: **name**, joka määrittelee parametrin nimen ja **value**, joka määrittelee parametrin arvon.

PRE

Määrittelee esimuotoillun tekstikappaleen. Aloitus- ja lopetuskomentojen välissä oleva teksti näytetään selaimessa sellaisenaan - ylimääräisiä välilyöntejä ja rivinvaihtoja ei heitetä pois. Teksti esitetään selaimessa tasalevyisellä fontilla.

Esimerkki:

```
<pre>
Nimi        Puhelinnumero
Sirkka      123 4567
Sakari      765 4321
</pre>
```

Komennolle voi antaa attribuutin **width**, jossa määritellään kappaleen leveys merkkeinä. Useimmat selaimet eivät kuitenkaan huomioi tätä attribuuttia.

SAMP

Määrittelee aloitus- ja lopetuskomentojen välissä olevan tekstin esimerkiksi ohjelman tai vastaavan tulosteista. Esimerkki:

```
Kun saat näytöllesi tekstin <samp>Login:</samp>, voit
kirjoittaa käyttäjätunnuksesi.
```

SCRIPT

Aloitus- ja lopetuskomentojen välissä oleva teksti tulkitaan komennoiksi. Jos selain tuntee käytetyn kielen, se ajaa nämä komennot. Attribuutilla **language** voidaan määritellä käytetty kieli.

SELECT

Määrittelee valikon. Komentoa voi käyttää vain **form**-komentojen välissä. Aloitus- ja lopetuskomentojen sijoitetaan **option**-komentoja, joilla merkitään valikon kohdat.

Komento voi saada seuraavia attribuutteja:

name	Kertoo valikon nimen. Tämä välitetään eteenpäin lomakkeen tietoja käsittelevälle ohjelmalle valitun kohdan arvon kera.
multiple	Kertoo, että valikosta voidaan valita useampia kohtia yhtäaikaa.
size	Jos komennossa on käytetty **multiple**-attribuuttia, kertoo miten monta valikon kohtaa on yhtä aikaa näkyvissä selaimen näytöllä.

SMALL

Aloitus- ja lopetuskomentojen välissä oleva teksti tulostetaan selaimen ruudulle normaalia pienemmällä tekstityypillä.

```
<small>Sivujen tekijä ei ota vastuuta sivujen
aiheuttamista vahingoista</small>
```

SPAN

Aloitus- ja lopetuskomennolla voidaan ympäröidä pala tekstiä, johon halutaan määritellä tiettyjä tyylimäärittelyitä. Tyylimäärittelyitä voidaan antaa esimerkiksi **style**-attribuutilla:

```
<p><span style="font-size: 5cm">T</span>estataan
todella isoa alkukirjainta.</p>
```

STRIKE

Aloitus- ja lopetuskomentojen välissä oleva teksti tulostetaan selaimen ruudulle yliviivattuna.

```
<p>Tämä elokuva on mielestäni <strike>täyttä
roskaa</strike> aika huono.
```

STRONG

Aloitus- ja lopetuskomentojen välissä oleva teksti tulostetaan selaimen ruudulle voimakkaasti korostettuna.

```
<p><strong>Älä missään tapauksessa</strong> yritä
herättää nukkuvaa karhua!
```

STYLE

Aloitus- ja lopetuskomentojen välissä määritellään sivun ulkoasuun vaikuttavia tyylikomentoja. Komennot sijoitetaan **head**-komentojen väliin.

SUB

Aloitus- ja lopetuskomentojen välissä oleva teksti määritellään näkymään selaimessa alaindeksinä. Tätä voidaan käyttää esimerkiksi matemaattisten tekstien yhteydessä.

```
<p>x<sub>1</sub> + x<sub>2</sub>
```

SUP

Aloitus- ja lopetuskomentojen välissä oleva teksti määritellään yläindeksiksi.

```
<p>Talon pinta-ala on 129 m<sup>2</sup>
```

TABLE

Aloittaa taulukon. Taulukon aloitus- ja lopetuskomentojen väliin sijoitetaan rivejä ja niiden sisään soluja.

```
<table>
<tr><th>Etunimi</th><th>Sukunimi</th></tr>
<tr><td>Aku</td><td>Ankka</td></tr>
<tr><td>Hessu</td><td>Hopo</td></tr>
<tr><td>Mikki</td><td>Hiiri</td></tr>
</table>
```

Taulukon aloituskomento voi saada seuraavia attribuutteja:

align	Määrittelee taulukon sijoituksen ruudulla. Voi saada arvot **left**, **center** tai **right**.
width	Määrittelee taulukon leveyden. Leveys voidaan antaa joko prosenttimääränä selaimen ikkunan leveydestä tai pikselimääränä. `<table width="100%">`
border	Määrittelee taulukon reunan paksuuden pikseleinä. Arvo **0** tekee taulukon, jonka reunat eivät näy ollenkaan.
cellspacing	Määrittelee taulukon solujen väliin tulevan tyhjän tilan määrän pikseleinä.
cellpadding	Määrittelee taulukon solujen sisään, tekstin ja solun reunan väliin, tulevan tyhjän tilan määrän pikseleinä.

TD

Määrittelee taulukon solun. Aloitus- ja lopetuskomentojen väliin sijoitetaan solun sisään tulevia HTML-komentoja. Solukomennon voi sijoittaa vain taulukon rivin sisään.

```
<table>
<tr><th>Etunimi</th><th>Sukunimi</th></tr>
<tr><td>Aku</td><td>Ankka</td></tr>
<tr><td>Hessu</td><td>Hopo</td></tr>
<tr><td>Mikki</td><td>Hiiri</td></tr>
</table>
```

Td-komento voi saada seuraavia attribuutteja:

nowrap	Määrittelee, että solun sisällön tulee säilyä yhdellä rivillä.
rowspan	Venyttää solun sisällön annetun rivimäärän kokoiseksi.
colspan	Venyttää solun sisällön annetun sarakemäärän kokoiseksi.
align	Määrittelee solun sisällön sijoittelun vaakasuunnassa. Vaihtoehdot ovat **left** (vasen reuna), **center** (keskitetty) ja **right** (oikea reuna). Jos tämä määrittely on annettu **tr**-komennossa, solun komennot ylittävät riville annetut määrittelyt.
valign	Määrittelee solun sisällön sijoittelun pystysuunnassa. Vaihtoehdot ovat **top** (yläreuna), **middle** (keskikohta) ja **bottom** (alareuna). Jos tämä määrittely on annettu **tr**-komennossa, solun komennot ylittävät riville annetut määrittelyt.
width	Solun leveys pikseleinä tai prosenttiarvona taulukon leveydestä. Selain pyrkii käyttämään tätä arvoa mahdollisuuksien mukaan, mutta jos pyydetty leveys ei ole mahdollinen, selain käyttää itse laskemaansa leveyttä.
height	Solun korkeus. Määrittely kuten **width**-attribuutissa.

TEXTAREA

Tekee lomakkeen sisään suuren tekstikentän. Komento voidaan sijoittaa vain **form**-komentojen väliin. Sille annetaan attribuutit **name** (joka välitetään lomakkeen tietoja käsittelevälle ohjelmalle kentän nimenä) sekä **cols** ja **rows** (jotka määrittelevät kentän koon sarakkeina ja riveinä).

Aloitus- ja lopetuskomennon väliin sijoitettu teksti tulee näkyviin tekstialueelle oletusarvoisena tekstinä.

```
<textarea name="kommentit" rows=4 cols=40>
</textare>
```

TH

Taulukon otsikkosolu. Komento toimii samalla tavalla kuin komento **td**, mutta sen sisältö näkyy taulukossa yleensä lihavoituna ja keskitettynä. Komento on tarkoitettu taulukon rivien ja sarakkeiden otsikoiden tekemiseen.

TITLE

Aloitus- ja lopetuskomennon välissä annetaan WWW-sivun nimi. Se tulee näkyviin selaimessa yleensä ikkunan yläreunassa. Lisäksi nimi näkyy esimerkiksi selaimen kirjainmerkeissä ja historiavalikossa. Myös monet hakukoneet käyttävät sitä hakutulosten otsikoissa.

Komento sijoitetaan **head**-komentojen väliin.

```
<html>
<head>
<title>Kotisivu</title>
</head>
<body>
<h1>Tervetuloa kotisivulleni!</h1>
</body>
</html>
```

TR

Määrittelee taulukon rivin. Komento sijoitetaan **table**-komentojen väliin. Komennon sisään voidaan sijoittaa **td**- tai **th**-komentoja. Komentoon kuuluu myös lopetuskomento.

```
<table>
<tr><th>Etunimi</th><th>Sukunimi</th></tr>
<tr><td>Aku</td><td>Ankka</td></tr>
<tr><td>Hessu</td><td>Hopo</td></tr>
<tr><td>Mikki</td><td>Hiiri</td></tr>
</table>
```

Tr-komento voi saada attribuutit **align** ja **valign**, jotka määritellään samalla tavalla kuin **td**-komennossa. Ne vaikuttavat kaikkiin niihin rivin soluihin, joiden **td**- tai **th**-komennossa ei ole annettu näitä määrittelyitä.

TT

Aloitus- ja lopetuskomentojen välissä oleva teksti tulostetaan selaimessa tasalevyisellä tekstityypillä (yleensä **Courier**-tyyppisenä).

```
<tt>Courier</tt>
```

U

Aloitus- ja lopetuskomentojen välissä oleva teksti alleviivataan.

UL

Tekee ranskalaiset viivat -listan. Aloitus- ja lopetuskomentojen väliin sijoitetaan listan kohtia **li**-komennolla.

```
<ul>
<li>Appelsiineja
<li>Mansikoita
<li>Perunoita
</ul>
```

Listan ulkoasua voidaan muokata attribuutilla **type**, joka voi saada arvot **disc** (listan kohtien eteen tulee pallo), **square** (kohtien eteen tulee neliö) tai **circle** (listan kohtien eteen tulee ympyrä). Attribuutti **compact** kertoo selaimelle, että lista tulisi näyttää mahdollisimman tiiviissä muodossa.

VAR

Aloitus- ja lopetuskomennon välissä oleva teksti määritellään muuttujaksi.

```
Sähköpostiosoitteesi on <var>tunnuksesi</var>@kone.fi.
```

WBR

Katso komentoa **nobr**.

MERKKIKOODIT

Jos haluat kirjoittaa WWW-sivullesi merkkejä, joita ei voi kirjoittaa näppäimistöltä, voit käyttää HTML-entiteettejä. Merkit voidaan kirjoittaa myös käyttäen entiteettiä, joka koostuu kirjaimista **&#**, merkin numeroarvosta ja puolipisteestä.

Seuraavilla sivuilla olevista taulukoista löydät listan erikoismerkeistä, entiteeteistä, joilla kirjain saadaan tulostettua WWW-sivulle ja vastaavista numerokomennoista. Huomaa, että entiteeteissä isojen ja pienten kirjainten erolla on merkitystä.

Alla lueteltujen entiteettien lisäksi voit käyttää seuraavia komentoja:

- **<** Pienempi kuin -merkki (<)
- **>** Suurempi kuin -merkki (>)
- **&** Et-merkki (&)
- ** ** Kiinteä välilyönti

Merkki	Komento	Entiteetti	Merkki	Komento	Entiteetti
¡	¡	¡	«	«	«
¢	&#162	¢	¬	¬	¬
£	&#163	£	-		
¤	¤	¤	®	®	®
¥	&#165	¥	¯	¯	¯
¦	&#166	¦	°	°	°
§	§	§	±	±	±
¨	¨	¨	2	²	²
©	©	©	3	³	³
a	ª	ª	´	´	´

Merkki	Komento	Entiteetti	Merkki	Komento	Entiteetti
µ	µ	µ	¿	¿	
¶	¶	¶	À	À	À
·	·	·	Á	Á	Á
¸	¸	¸	Â	Â	ˆ
¹	¹	¹	Ã	Ã	Ã
º	º	º	Ä	Ä	Ä
»	»	»	Å	Å	Å
¼	¼	¼	Æ	Æ	Æ
½	½	½	Ç	Ç	Ç
¾	¾	¾	È	È	È
			É	É	É

Merkki	Komento	Entiteetti	Merkki	Komento	Entiteetti
Ê	Ê	Ê	Õ	Õ	Õ
Ë	Ë	Ë	Ö	Ö	Ö
Ì	Ì	Ì	×	×	
Í	Í	Í	Ø	Ø	Ø
Î	Î	Î	Ù	Ù	Ù
Ï	Ï	Ï	Ú	Ú	Ú
Ð	Ð	Ð	Û	Û	Û
Ñ	Ñ	Ñ	Ü	Ü	Ü
Ò	Ò	Ò	Ý	Ý	Ý
Ó	Ó	Ó	Þ	Þ	Þ
Ô	Ô	Ô	ß	ß	ß

Merkki	Komento	Entiteetti	Merkki	Komento	Entiteetti
à	à	à	ë	ë	ë
á	á	á	ì	ì	ì
â	&#226	â	í	í	í
ã	ã	ã	î	î	î
ä	ä	ä	ï	ï	ï
å	å	å	ð	ð	&ieth;
æ	æ	æ	ñ	ñ	ñ
ç	ç	ç	ò	ò	ò
è	è	è	ó	ó	ó
é	é	é	ô	ô	ô
ê	ê	ê	õ	õ	õ

Merk-ki	Komento	Entiteetti
ö	ö	ö
÷	÷	
ø	ø	ø
ù	ù	ù
ú	ú	ú
û	û	û
ü	ü	ü
ý	ý	ý
þ	þ	þ
ÿ	ÿ	ÿ

LÄHTEET JA LINKIT

Alta löydät linkkejä tässä kirjassa käytettyihin lähdemateriaaleihin ja muihinkin hyödyllisiin WWW-sivujen suunnittelua käsitteleviin sivuihin.

A Web Site is a Harsh Mistress
Diane Wilsonin artikkeli sivujensuunnittelusta.

```
http://www.lava.net/%7edewilson/web/rant.html
```

Alertbox
Jakob Nielsenin kaksi kertaa kuukaudessa ilmestyvä kolumni WWW-suunnittelusta.

```
http://www.useit.com/alertbox
```

Builder.com
Kattava kokoelma artikkeleita ja vihjeitä WWW-sivujen suunnittelijoille.

```
http://www.builder.com/
```

Creating Net Sites
Netscapen sivuilla oleva ohje WWW-sivujen luomiseen.

```
http://www.netscape.com/assist/net_sites/index.html
```

Dynamic Layers in Netscape
Ohjeita ja esimerkkejä tyylitiedostoista ja tasojen määrittelyistä Netscapessa.

```
http://developer.netscape.com/library/
documentation/communicator/dynhtml/index.htm
```

Elements of Web Design
Tietopaketti sivujen ulkoasun ja rakenteen suunnittelusta.

http://www.cnet.com/Content/Builder/Graphics/Design/

HTML tags sorted by Navigator releases
Listaa eri Netscape-versioissa toimivat HTML-komennot versionumeron mukaan.

http://developer.netscape.com/library/documentation/htmlguid/tags_complete.html

JavaWorld
Verkkolehti, joka sisältää laadukkaita artikkeleita Java-ohjelmoijille.

http://www.javaworld.com/

Learning HTML 3.2 by Examples
Kattava tietopaketti HTML-kielestä ja sen komennoista.

http://www.hut.fi/u/jkorpela/HTML3.2/

Lehtori
Suomalainen palvelu, jonka avulla voit tarkistaa sivuillasi käyttämäsi HTML-kielen. Ohjelma tarkistaa virheet HTML-tiedostostasi.

http://www.snowman.sci.fi/lehtori/

Netscape Advanced FAQ
Kysymyksiä ja vastauksia erikoisemmista sivujensuunnitteluun liittyvistä asioista.

http://developer.netscape.com/support/faqs/advhtml/advhtml.html

Netscape documentation
Kattava kokoelma tietoutta Netscape-selaimesta ja sen ymmärtämistä komennoista.

http://developer.netscape.com/library/documentation

NetscapeWorld
Verkkolehti, joka sisältää laadukkaita artikkeleita sivujensuunnittelusta ja ohjelmoinnista.

```
http://www.netscapeworld.com/
```

The Background Sampler
Netscapen sivuilla oleva kokoelma vapaasti käytettäviä taustakuvia WWW-sivuille.

```
http://www.netscape.com/assist/net_sites/bg/
backgrounds.html
```

The Internet Archive
Palvelu, jonka tavoitteena on säästää kaikki maailman WWW-sivut tuleville sukupolville.

```
http://www.archive.org/
```

The Web Techniques Perl Columns
Vinkkejä ja ohjeita CGI-ohjelmoinnista **Perl**-kielellä.

```
http://w3.stonehenge.com/merlyn/WebTechniques/
```

W3C - World Wide Web Consortium
Standardeja liittyen HTML:ään ja WWW:hen.

```
http://www.w3.org/
```

Web Page Backward Compatibility Viewer
Palvelu, jonka avulla voit katsella sivujasi sellaisina kuin ne näyttävät vanhemmissa selaimissa.

```
http://www.delorie.com/web/wpbcv.html
```

Webreference.com - The Webmaster's Library
Kattava kokoelma ohjeita ja vinkkejä WWW-sivujen suunnittelijoille.

http://www.webreference.com/

WWW Viewer Test Page
Sivu, joka sisältää tietoa selaimien apuohjelmista. Tällä sivulla voit myös testata erityyppisiä tiedostoja ja katsoa, miten oma selaimesi osaa näyttää ne.

http://www-dsed.llnl.gov/documents/WWWtest.html

Yahoo: World Wide Web
Laaja kokoelma linkkejä kaikista WWW:hen liittyvistä aiheista.

http://www.yahoo.com/Computers/Internet/World_Wide_Web/

Yale C/AIM Web Style Guide
Tyyliopas WWW-sivujen suunnittelijoille.

http://info.med.yale.edu/caim/manual

OHJELMAT

Alta löydät listan tekstissä mainituista ohjelmista ja osoitteet, josta voit kopioida ohjelmat itsellesi.

Netscape Communicator	**http://www.netscape.com**
Internet Explorer	**http://www.microsoft.com**
HTMLpad	**http://www.intermania.com**
Paint Shop Pro	**http://www.jasc.com**
WS-FTP	**http://www.ipswitch.com/Products/WS_FTP**
Fetch (Macintosh)	**http://www.dartmouth.edu/pages/softdev/fetch.html**
Color Manipulation Device	**http://www.meat.com/software/cmd/**
GIF Construction Set	**http://www.mindworkshop.com/alchemy/gifcon.html**
LiveImage	**http://www.mediatec.com**
mailto:formatter	**http://junior.apk.net/~jbarta/tutor/forms/mtfsites.html**
Astra Sitemanager	**http://www.merc-int.com**
Quicktime-plug-in	**http://quicktime.apple.com**
Shockwave	**http://www.macromedia.com**
Flash	**http://www.macromedia.com**

HAKEMISTO

A

A, 70, 335
  href, 70, 336
  name, 335
  target, 229, 336
above, 302
ADDRESS, 135, 337
aiff, 266
AIFF, 322
alaindeksi, 40
alihakemistot, 162
Alku, 216
alleviivaus, 39
aloituskomento, 28, 322
Alta Vista, 217
Analog, 213
animaatio, 147
applet, 322
APPLET, 337
  align, 338
  alt, 337
  code, 337
  codebase, 337
  height, 337
  hspace, 338
  name, 338
  vspace, 338
  width, 337
apuohjelma, 322
apuohjelmat, 19
AREA, 156, 338
  alt, 338
  coords, 156, 338
  href, 156, 338
  nohref, 158, 338
  shape, 156, 338
Ascii, 87
ASCII, 322
asiakasohjelma, 322
assosiatiiviset muuttujat, 310
Astra Sitemanager, 213
Attribute, 322
attribuutit, 29, 335
attribuutti, 322
au, 266
AU, 323
avi, 267
AVI, 323

B

B, 39, 338
background, 302
background (tyylikomento), 241
background-color, 256
background-image, 256
Back-painike, 20
BASE, 339
  href, 339
  target, 340
BASEFONT, 340
  size, 107, 108, 340
below, 302
bgColor, 302
BGSOUND, 265, 340
  loop, 265
  src, 265
BIG, 39, 40, 340
Binary, 87
BLINK, 340
BLOCKQUOTE, 136, 341
BODY, 79, 341

alink, 99, 342
background, 105, 342
bgcolor, 98, 341
link, 99, 341
text, 98, 341
vlink, 99, 341
boottaaminen, 323
BR, 33, 342
clear, 63, 342
Browse, 35
Browser, 332

C

CAPTION, 125, 342
align, 125, 342
Cardinal Company Index, 216
Cascading Style Sheet, 230
Cell, 332
CENTER, 343
CGI, 323
CGI-hakemisto, 305
CGI-ohjelma, 323
CGI-ohjelmat, 305
esimerkki, 306
parametrit, 307
checkbox, 176
chmod, 95
Choose File, 36
CITE, 41, 137, 343
Client, 322
Clinet, 92
clip, 301
clip (tyylikomento), 255
CNET, 190
CNN, 189
CODE, 42, 343
color (tyylikomento), 241
Color Manipulation Device, 103
Compart, 92
Constructor, 326
Copy, 23
Courier, 130
Courier-kirjasintyyli, 39
Cute Communications, 92

D

DD, 134, 343
debuggaus, 206
DFN, 42, 344
DIR, 344
DIV, 236, 344
align, 344
style, 236
DL, 134, 345
compact, 345
DLC, 93
document, 301
document.write, 270
DT, 134, 345

E

editori, 12
EM, 42, 345
EMBED, 265, 267, 345
align, 266, 346
autostart, 266
border, 346
frameborder, 346
height, 266, 346
hidden, 266, 346
hspace, 346
name, 346
pluginspace, 268, 346
src, 265, 346
type, 346
vspace, 346
width, 266, 346
entiteetit, 372
entiteetti, 46, 323
erikoiskappaleet, 130
esimerkki (tekstityyli), 42
esimuotoiltu teksti, 130
etusivu, 191
graafinen, 191
tekstipohjainen, 194
tekstivaihtoehto, 193
etusivutiedosto, 91, 167
Eunet, 93
Event Handler, 332
Excite, 217
exe, 269
External viewer, 322
extranet, 215
Extranet, 323

F

FAQ, 210
Fetch, 19, 84, 323
flash, 268
Flash, 323
FONT, 346
  color, 101, 347
  face, 109, 347
  size, 107, 347
font-family, 240
fontit, 109
font-size, 240
font-style, 240
font-weight, 240
FORM, 170, 347
  action, 170, 309, 347
  enctype, 348
  method, 170, 348
  name, 289
formaatti, 323
Forward-painike, 20
frame, 326. *Katso* kehykset
FRAME, 222, 348
  frameborder, 349
  marginheight, 228, 260, 349
  marginwidth, 228, 260, 349
  name, 228, 348
  noresize, 227, 349
  scrolling, 227, 349
  src, 224, 348
FRAMESET, 220, 349
  border, 260
  cols, 221, 350
  frameborder, 260
  framepadding, 260
  framespacing, 260
  rows, 221, 349
Frontpage, 17
Frontpage Express, 16
Ftp, 84
FTP, 18, 324
Function, 324
funktio, 324
fyysinen, 324

G

gif, 48, 52, 138, 143
  animaatio, 147
  interlace, 144
  värimäärät, 145
Gif Construction Set, 147, 149
Gopher, 324
graafinen etusivu, 191

H

H1, 33, 350
  align, 34, 350
hakemistorakenne, 166
hakemistot, 162
hakemistot ja osoitteet, 164
hakukoneet, 217
hakuohjelma, 324
HEAD, 77, 350
height (tyylikomento), 255
heksadesimaalit, 324
hierarkkinen rakenne, 187
  suositeltava syvyys ja leveys, 188
historiavalikko, 78
Hotbot, 218
HR, 43, 351
  align, 44, 351
  noshade, 351
  size, 44, 351
  width, 44, 351
htm, 24
HTML, 28, 77, 324, 351
  komento, 28
  sisentäminen, 31
  version, 352
HTML 3.2, 183, 379
HTML 4.0, 183
HTML-editori, 15
HTML-komento, 325
HTMLpad, 15
HTML-tiedosto, 325
http, 325
Hypermedia, 325
hyperteksti, 70
Hyperteksti, 325
Hypertekstipalvelin, 334

I

I, 39, 352
IFRAME, 352

ILAYER, 252, 353
IMG, 53, 353
  align, 353
  alt, 57, 353
  border, 354
  height, 203, 261, 354
  hspace, 263, 354
  ismap, 354
  name, 289
  src, 53, 353
  usemap, 158, 354
  vspace, 263, 354
  width, 203, 261, 354
import, 232
imuroitavat ohjelmat, 269
index.html, 92, 167
Infoseek, 218
INPUT, 171
  align, 356
  checked, 355
  maxlength, 174, 356
  name, 355
  size, 174, 355
  src, 356
  type, 171, 176, 354
  value, 175, 355
INPUT name, 171
interaktiivinen kuva, 325
interlace, 144
Interlace, 325
Internet Explorer, 11, 325
Internet-operaattori, 83, 325
intranet, 214, 318
Intranet, 325
ISINDEX, 356
  prompt, 356
ISO-8859-1, 326
IWN, 93

J

Japo, 93
Java, 326
JavaScript, 270, 326
  aikojen käsittely, 284
  alert, 298
  Array-olio, 284
  attribuutit, 282
  Boolean-olio, 287
  Boolen arvot, 276
  Date-olio, 284
  document-olio, 287
  ehtolause, 277
  else, 278
  entiteetit, 276
  erikoismerkit, 275
  for, 279
  Function-olio, 287
  funktion palautusarvot, 281
  funktion parametrit, 280
  funktiot, 280
  history-olio, 295
  if, 277
  ikkunan avaaminen, 291
  ja tasot, 300
  kommentit, 271
  konstruktori, 282
  kontrollikomennot, 277
  lainausmerkit, 272
  laskutoimitukset, 274
  lisätietoja, 304
  location-olio, 294
  loogiset operaattorit, 278
  matemaattiset operaattorit, 285
  Math-olio, 285
  merkkijonot, 275, 286
  metodit, 282
  muuttujat, 273
  navigator-olio, 296
  new, 282
  Number-olio, 287
  oliot, 282
  onMouseOver, 297
  return, 281
  silmukat, 278, 279
  String-olio, 286
  tapahtumankäsittelijät, 297
  tapahtumat, 297
  taulukot, 284
  this, 282
  tulostaminen, 270
  var, 273
  varmistusikkuna, 293
  virheet, 272
  while, 278
  window-olio, 290
JavaWorld, 379
jpeg, 48, 52, 139, 203

pakkausmäärä, 140
progressive, 145
värimäärät, 145
julkistaminen, 216

K

karttatiedosto, 154, 156
apuohjelmia, 159
monikulmio, 157
nelikulmio, 156
tausta, 158
ympyrä, 157
KBD, 42, 356
kehykset, 220
koon muuttaminen, 227
linkkien toiminta, 228
marginaalit, 228
määrittely, 220
sisällön määrittely, 222
vierityspalkit, 227
kehys, 326
Keskusteluryhmä, 333
kirjanmerkit, 78
Kirjanmerkit, 326
kirjasintyypin vaihtaminen, 109
kirjasintyypit
fontti, 109
tekstin koko, 107
tekstin väri, 98
kirjoituskonetyyli, 39
kirjoitustyyli, 313
klikattava kuva, 325
kohderyhmä, 317
kokonainen HTML-tiedosto, 80
Kolumbus, 93
kommentit, 75
konstruktori, 326
koodi, 326
koodi (tekstimuotoilu), 42
korostus, 42
kotihakemisto, 327
kotisivu, 327
kotisivuluettelo, 186
kotisivun osoite, 89
kursiivi, 39
kuuma alue, 327
kuuma sana, 70, 327
kuuma teksti, 327
kuva
animaatio, 147
interaktiivinen kuva, 153
koko, 50
koordinaatit, 155
kuuma kohta, 154, 155
kuvankäsittely, 138
kuvien lataaminen etukäteen, 205
läpinäkyvä väri, 141
osoitteen selvittäminen, 66
paloitellut kuvat, 257
postimerkkikuva, 160
rikkinäiset kuvat, 55
selitysteksti, 56
siirtonopeus, 146
sijoittaminen WWW-sivulle, 53
sijottelu, 59
tallentaminen, 52, 138
tallentaminen WWW-sivulta, 68
taustaväri, 50
tekstin kierrättäminen kuvan ympäri, 62
työskentely kuvien kanssa, 58
venyttäminen, 205
verkossa olevan kuvan liittäminen, 64
vähitellen näkyviin tuleva, 144
värimäärä, 50
värimäärän muuttaminen, 146
värimäärät, 145
kuvakartta, 154, 156
kuvankäsittelyohjelma, 17, 48
kuvaruudulta lukeminen, 313
kuvien lataus, 56
käyntikorttisivut, 317
käyttäjätunnus, 83, 85
käyttöliittymä, 199
käytön seuranta, 211

L

lainaukset, 136
layer, 332
LAYER, 243, 356
above, 250, 357
background, 245, 357
below, 250, 357
bgcolor, 245, 357
clip, 248, 357

height, 243, 356
id, 245, 357
left, 243, 357
pagex, 246, 357
pagey, 246, 357
src, 247, 357
top, 243, 357
visibility, 252, 357
width, 243, 356
z-order, 250, 357
layout, 98, 220, 243, 257
pikselikuvien käyttö, 261
left, 301
left (tyylikomento), 255
leikepöytä, 23
leikkaaminen ja liimaaminen, 23
leipäteksti, 29
keskitetyt kappaleet, 32
oikean reunan tasaus, 32
Rivinvaihdot, 33
sisentäminen, 31
tavutus, 31
tyhjän tilan tekeminen, 31
LI, 132, 133, 358
type, 358
value, 358
lihavointi, 39
line-height, 241
LINK, 358
href, 359
rel, 233, 358
rev, 359
title, 359
type, 233, 359
Linka, 216
linkin väri, 99
linkit, 70, 72
kuvalinkki, 73
toisiin sivuihin, 70
valitseminen, 70
linkit edelliseen hakemistoon, 163
linkit hakemistojen välillä, 162
linkki, 327
linkkien tarkistus, 209
ohjelmia, 209
Lisko, 216
listat, 132
määritelmälistat, 134
numeroidut listat, 132
ranskalaiset viivat, 133
LiveImage, 159
load, 303
logien seurantatyökalut, 213
logitiedostot, 211
loki, 327
lomake, 327
käsittelyohjelma, 170
lomakealue, 170
painikkeet, 176
piilotetut kentät, 176
radiopainike, 176
rasti ruutuun -painike, 176
tekstialueet, 175
tekstikentät, 171
tekstilaatikot, 171
valikot, 180
lomakkeet, 170
CGI-ohjelmat, 309
tietojen purkaminen CGI-ohjelmalla, 309
looginen, 327
loogiset muotoilukomennot, 41
lopetuskomento, 28, 40, 327
lukunopeus, 313
lukuoikeudet, 93, 328
Lycos, 218
Lynx, 11, 328
läpinäkyvät värit, 141

M

Macromind, 268
mailto (URL), 210
Mailto-formatter, 173
MAP, 156, 359
name, 156, 359
margin- (tyylikomento), 242
mato, 324
matomerkki, 89
Megabaud, 93
MENU, 359
compact, 359
META, 360
content, 196, 360
http-equiv, 196, 360
name, 360
Method, 328
metodi, 328

Microsoft Word, 17
mid, 266
midi-tiedostot, 266
moveAbove, 303
moveBy, 302
moveTo, 302
moveToAbsolute, 302
Mozilla, 328
multimedia, 265
  ääni, 265
muuttuja, 328
muuttuja (tekstityyli), 43
määritelmälistat, 134

N

name, 301
navigoinnin ulkoasu, 199
navigointi, 185
navigointilinkit, 200
navigointipalkki, 199
Netscape, 10, 328, 379
  Composer, 16
NetscapeWorld, 380
Netti Finland, 93
NOBR, 360
NOFRAMES, 220, 361
NOLAYER, 252, 362
nopeus, 202
NOSCRIPT, 362
Notepad, 12, 13
  käyttäminen, 22
numeroidut listat, 132
näppäilyt (tekstityyppi), 42

O

Object, 328
ohjelmien välillä siirtyminen, 25
OL, 132, 362
  compact, 363
  start, 363
  type, 362
olio, 328
Open File, 35
Opera, 11
operaattori, 83, 96, 168, 328
operaattorien ohjesivuja, 92
OPTION, 180, 363
  selected, 182, 363
  value, 181, 363
orvot sivut, 198
osoite, 20
osoitepalkki, 21
osoitteet, 135
otsikkoalue, 329
otsikkotiedot, 77
otsikot, 33
  keskitetyt otsikot, 34
  koko, 34

P

P, 29, 363
  align, 32, 363
Paint Shop Pro, 18, 48, 329
  interlace, 144
  koordinaattien etsiminen, 156
  kuvan koon pienentäminen, 202
  kuvan värimäärä, 146
  läpinäkyvän värin tekeminen, 142
  piirtotyökalut, 51
  progressive jpeg, 145
  värimäärän vähentäminen, 203
  väripaletti, 51
pakatut ohjelmat, 269
palaute, 209
palautearvo, 329
palautelomake, 172
palvelin, 305, 329
palvelinkone, 83, 85, 96, 165, 329
  osoite, 83
PARAM, 364
parametri, 329
parentLayer, 302
Password-kenttä, 175
Paste, 23
PCUF, 93
Perl, 306, 310, 329, 380
peräkkäiset sivut, 186
Photoshop, 18, 48, 329
pieni teksti, 40
pienoismalli sivuista, 312
piirtäminen, 49
pikseli, 155, 330
pikselikuvat, 261
  läpinäkyvät pikselit, 262
plug-in, 267, 330

polku, 330
position, 253
postimerkki, 330
postimerkkikuvat, 160
PRE, 130, 364
  width, 364
progressiivinen jpeg, 330
protokolla, 330
public_html, 85, 95
puumainen rakenne, 187
pääteyhteys, 330

Q

quicktime, 267
Quicktime, 331

R

radiopainike, 176
ranskalaiset viivat, 133
Reload, 331
Reload-painike, 20, 36, 58
resizeBy, 302
resizeTo, 302
Resoluutio, 331
Return Value, 329
RGB-arvo, 102, 331
rivinvaihdot, 130
rivinvaihto, 29, 33
robotti, 324
runko, 331

S

salasana, 83, 85
salasanasuojaus, 214
SAMP, 42, 364
sanahaku, 190
sanomalehti, 121
Sci.fi, 93
SCRIPT, 270, 365
  language, 270
selain, 9, 332
  käyttäminen, 20
  sivujen katseleminen, 35
  tekstiselain, 58
  välimuisti, 207
SELECT, 180, 365
  multiple, 182, 365
  name, 180, 365
  size, 365
selitysteksti, 332
selkeys, 202
Serveri, 216
shockwave, 268
Shockwave, 332
siblingAbove, 302
siblingBelow, 302
silmäiltävyys, 314
SIP, 93
sisällysluettelo, 186
sitaatit, 41
Sitemill, 17
sivu
  värit, 98
sivujen katseleminen, 35
sivujen käyttötarkoitus, 317
sivujen nimet, 197
sivujen suunnittelu, 312
sivujen toiminnan testaaminen, 206
sivukokonaisuuksien suunnittelu, 184
sivun runko, 79
sivun ulkoasu, 230
sivut
  erikoisefektit, 184
  erityyliset sivut, 184
  julkaiseminen, 83
  muiden tekemien sivujen katselu, 81
  navigointi, 185
  osoite, 89
  pohja, 80
  päivittäminen, 83, 208
  rakenne, 76, 185
  runko, 79
  siirtäminen palvelinkoneelle, 83
  sivun nimi, 77
  tarkoitus, 184
  yleisö, 184
SMALL, 39, 40, 365
solu, 332
source-include, 255
SPAN, 237, 365
  style, 237, 365
src, 302
Stop-painike, 20

STRIKE, 39, 40, 366
STRONG, 42, 366
STYLE, 234, 366
  type, 234
style sheet, 230
stylesheet, 233
SUB, 39, 40, 366
Submit-painike, 172
SUP, 39, 40, 366
suuri teksti, 40

T

TABLE, 111, 367
  align, 367
  background, 127
  bgcolor, 126
  border, 112, 118, 367
  cellpadding, 117, 367
  cellspacing, 117, 367
  height, 114
  width, 114, 367
Tab-näppäin, 26, 36
tallentaminen, 23, 35
tapahtumankäsittelijä, 332
tarkennin
  exe, 168, 269
  gif, 168
  htm, 168
  html, 168
  jpeg, 168
  jpg, 168
  txt, 168
  zip, 269
taso, 332
tasot, 243, 378
  HTML-komennoilla, 243
  JavaScript-olio, 300
  järjestys, 250
  leikkaaminen, 248
  olioiden attribuutit, 301
  sisäkkäin sijoittelu, 245
  sisältö, 247
  tapahtumankäsittelijät, 303
  tason metodit, 302
  taso-olion luominen, 303
  tasot ja JavaScript, 300
  tyylimäärittelyiden avulla, 253
taulukot, 111, 130
  koko, 114
  korkeus, 111
  käyttö layoutin suunnitelussa, 259
  käytännössä, 127
  leveys, 111
  otsikkosolut, 123
  otsikkoteksti, 125
  rivien yhdistäminen, 120
  rivit, 112
  sarakkeet, 111
  solujen koko, 115
  solujen sisällön sijainti, 118
  solujen taustakuva, 127
  solujen taustaväri, 259
  solujen väritys, 126
  solujen yhdistäminen, 120
  solut, 113
  tyhjät solut, 119
  virheiden etsiminen, 128
taustakuvat, 105
taustan väri, 98
tavutus, 31
TD, 113, 367
  align, 118, 368
  background, 127
  bgcolor, 126
  colspan, 120, 368
  height, 115, 368
  nowrap, 119, 368
  rowspan, 120, 368
  valign, 118, 368
  width, 115, 116, 368
tekijänoikeudet, 68
tekstieditori, 13
  käyttäminen, 22
tekstikappaleet, 29
tekstilaatikko, 332
tekstin jakaminen paloiksi, 315
tekstin koko, 107
tekstin väri, 98
tekstinkäsittelyohjelma, 13
tekstipohjainen etusivu, 194
Teksturi, 12
Tele iNet Pro, 93
Telnet, 330
termien määrittely (tekstimuotoilu), 42
tervetulotoivotus-sivu, 195
TEXTAREA, 175, 368

cols, 175, 368
name, 175, 368
rows, 175, 368
text-decoration, 241
text-indent, 242
TH, 123, 369
align, 123
background, 127
thumbnail, 160
tiedostoarkistot, 332
tiedostojen siirto, 86
siirtomuoto, 87
tiedostokoko, 202
tiedostonimen tarkennin, 332
tiedostonimet
tarkennin, 24, 35, 52, 138, 139, 168
tiedostot, 162
työskentelytavat, 95
tiedostotarkennin .css, 232
tiedottaminen, 216
TITLE, 77, 197, 369
top (tyylikomento), 255
TR, 112, 369
align, 118, 370
background, 127
bgcolor, 126
height, 115
valign, 118, 370
width, 115
TT, 39, 370
Tucows, 17, 104, 160, 209, 214
tyyliluokat, 238, 239
tyylimäärittely, 230, 333
tyylimäärittelyt, 230
fonttikoko, 232
HTML-komennoissa, 236
HTML-tiedostossa, 234
komentoja, 239
komentojen ulkoasun muuttaminen, 230
tasot, 253
yhdistäminen, 232
tyylipohja, 230, 333
tyylit
fyysiset, 38
loogiset, 38, 41
Yhdistäminen, 40
tyylitiedosto, 230
tyylitiedostot, 232, 312
työympäristö, 9
tärkeimmät asiat sivulla, 316
töiden hajauttaminen, 208

U

U, 39, 370
UL, 133, 370
ulkoasu, 199
ulkomaiset kirjaimet, 46
URL, 20, 72, 333
usein kysytyt kysymykset, 210
Uutisryhmä, 333
uutisryhmät
uutuudet, 218
uutuuslistat, 216

V

vahva korostus (tekstityyli), 42
VAR, 43, 370
Variable, 328
vasteaika, 206
verkkorakenne, 189
videoleikkeet, 266
View source, 333
View Source, 81
virheelliset komennot, 37
virheilmoitukset, 90, 94
visibility, 255, 301
vxtreme, 267
vähitellen näkyviin tuleva kuva, 144
välilyönti, 29, 31, 130
väliotsikot, 33, 314
väliviiva, 43
paksuus ja leveys, 44
värien kanssa työskentely, 104
värien käyttö, 202
värien nimet, 100
värimäärät, 145
värit, 98
määrittely, 102

W

W3C, 183
WakkaNet, 93
wav, 265, 266

WAV, 333
WBR, 361
Web, 334
Web Page Backward Compatibility Viewer, 207
Webmaster, 334
Webstudio, 216
width (tyylikomento), 255
Windows 95
  videotiedostoja, 267
  äänitiedostoja, 266
WS-FTP, 19, 84, 334
  asetukset, 84
  asetuksien tallentaminen, 86
  hakemistojen siirtäminen, 164
WWW, 334
WWW-palvelin, 334
WWW-sivut
  avaaminen, 35
WYSIWYG, 16
WYSIWYG-editori, 16

Y

Yahoo, 127, 216
yhdistelmärakenne, 190
yliviivattu teksti, 40
ylläpito, 206, 317
Ylläpitäjä, 334
yläindeksi, 40

Z

z-index, 255
zIndex, 301

Ä

ääkköset, 45
  tiedoston nimessä, 55